Kernel Dimension Reduction Approaches for Multivariate Process Control

Thrasivoulos Tsagaroulis

A Thesis

in

The Concordia Institute

for

Information Systems Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Master of Applied Science (Quality Systems Engineering) at

Concordia University

Montréal, Québec, Canada

December 2007

Your file Votre référence
ISBN: 978-0-494-40932-9
Our file Notre référence
ISBN: 978-0-494-40932-9

Abstract

Kernel Dimension Reduction Approaches for Multivariate Process Control

Thrasivoulos Tsagaroulis

The great challenge in quality control is to devise computationally efficient algorithms to detect and diagnose process defects. Univariate statistical process control charts are currently used as an integral part in statistical quality control of engineering processes. Unfortunately, most data are inherently multivariate and need to be modelled accordingly. Major limitations such as higher data complexity and difficulty in interpretation have limited the application of multivariate techniques in process control. Motivated by the recent advances in dimensionality reduction algorithms and in order to effectively monitor highly correlated data, we introduce in this thesis new multivariate statistical process control charts based on the eigen-analysis of kernel matrices. The core idea behind our proposed techniques is to develop a theoretically rigorous methodology for multivariate statistical process control. We use scalp-recorded electroencephalograms (EEGs) as our real-world multivariate data source to demonstrate the effectiveness of our proposed algorithms. EEGs consist of vast amounts of complex data that require a trained professional to perform a proper analysis. Moreover, the currently used methodologies for analyzing EEGs are very labor-intensive. To circumvent these limitations, we show through extensive experimentation that our proposed approaches can be applied successfully in the analysis of EEGs by automating the detection of events. The task of classifying the events would still, however, be left to a professional clinician.

For ease of visualization and analysis of EEGs, we designed a user-friendly Graphical User Interface (GUI) to test the performance of the proposed kernel dimension reduction techniques, and to also perform a comparison with the most prevalent methods used in multivariate process control.

Acknowledgements

I would like to express my sincere gratitude to my advisor, Dr. A. Ben Hamza, for his continuous support and encouragement throughout my graduate studies. His mentorship is essential to the completion of this thesis, more importantly, the challenging research that lies behind it.

I would also like to thank Dr. Rajeev Agarwal, for providing and reviewing the EEG data.

Finally, I would also like to thank my wonderful wife and our beautiful new daughter for their unconditional support and love during this process.

Table of Contents

List of Tables

List of Figures

CHAPTER 1

Introduction

The use of multivariate statistical process control is facilitated by the proliferation of sensor data that is typically complex, high-dimensional and generally correlated. Almost every new experiment or application requires the availability of considerable amount of data for the purpose of analysis. Depending on the type of data, various techniques of data analysis are used to extract useful information and arrive at conclusions. Modern industrial processes contain a large number of variables that are regularly monitored and inspected for defects [1].

This type of process monitoring is known as statistical process control (SPC). The concept is based on the assumption that high variation leads to inferior quality. Therefore when several process parameters are controlled within specific targets, the end product tends to be in control and within specification. To achieve this, samples are frequently collected for each variable and displayed with the visual help of a control chart [2]. An example would be the measurement of solder paste on Printed Circuit Board Assemblies (PCBA) when components are being soldered in temperature controlled ovens. The solder paste thickness is measured at several locations on a specific quantity of PCBAs at the beginning of a production run. If the thickness is within specification at the fixed locations, the run is given the go ahead. If multivariate techniques were used, the correlation between the measurements at the various locations on the various PCBAs could guide the maintenance technician or engineer in diagnosing the root cause of insufficient solder or too much solder - two of the main causes of failures for newly manufactured PCBAs.

A control chart is a useful statistical tool that can be used to distinguish and detect between common causes of variation (random noise) and special causes (signal). The samples are plotted over time between

two thresholds defined as control limits. Samples exceeding the limits indicate a fault or a special cause of variation. Any outliers indicate that the process needs to be investigated to determine and remove the root cause of variation [3]. Figure 1.1 depicts such a control chart.

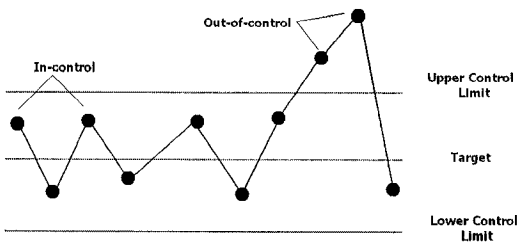

Figure 1.1: Control Chart.

Electroencephalographs (EEGs) are an example where vast amounts of complex data are present and require a trained professional many hours to perform analysis.

In this thesis, we present multivariate statistical dimensionality-reduction techniques for analyzing complex data.

1.1 Framework and Motivation

Univariate statistical process control is widely used to monitor and diagnose faults and outliers, as in the example of the solder paste measurement. For attribute variables, common charts used are the count chart (c chart) and the fraction defective chart (p chart). For continuous variables, operators and engineers use the the X-bar chart, R chart and S chart. These tools are very well documented and understood [5].

Unfortunately, variables are highly correlated in most processes. Currently, most industries model their processes by monitoring each variable independently of the other. The thickness of solder paste at the various locations is treated as a separate variable, for example, in the electronics manufacturing industry. As a consequence, operators are overwhelmed and have difficulty in determine the root cause of failure when there is insufficient solder or too much solder in several places during a single production run [5]. In Figure 1.2, two highly correlated variables are modelled dependently. The process is operating under normal operating conditions within the ellipse. Any sample falling outside the ellipse is an outlier. However, if the

variables were modelled independently, the control region would be defined within the rectangle. Some out-of-control observations would be misidentified as being in control.

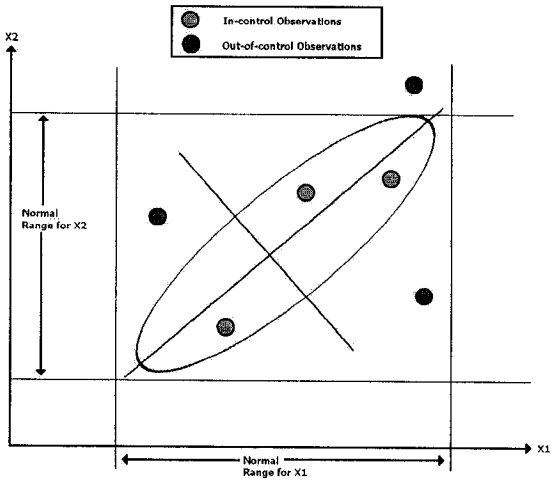

Figure 1.2: Covariance Between Variables.

1.2 Background

This thesis addresses the application of dimensionality reduction techniques for the analysis of multivariate data. The following background material is presented to provide context for this work.

1.2.1 Hotelling's T-squared Statistic

Let $X = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n]^T$ be an $n \times p$ data matrix of n vectors $\boldsymbol{x}_i \in \mathbb{R}^p$, where each observation $\boldsymbol{x}_i = (x_{i1}, \ldots, x_{ip})$ is a row vector with p variables.

Phase I of the T^2 control chart consists of establishing an outlier free reference sample [5]. Hotelling's T^2 statistic, also referred to as Mahalanobis distance, is defined as

$$T_i^2 = (\boldsymbol{x}_i - \bar{\boldsymbol{x}})S^{-1}(\boldsymbol{x}_i - \bar{\boldsymbol{x}})^T$$

where

$$\bar{\boldsymbol{x}} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i \quad \text{and} \quad S = \frac{1}{n-1}\sum_{i=1}^{n} (\boldsymbol{x}_i - \bar{\boldsymbol{x}})^T (\boldsymbol{x}_i - \bar{\boldsymbol{x}})$$

are the sample mean and covariance matrix respectively.

The Phase I upper control limit (UCL) and lower control limit (LCL) of T^2 control chart are given by

$$\begin{aligned} UCL &= \frac{(n-1)^2}{n} \frac{[\frac{p}{n-p-1}]F_{\frac{\alpha}{2},p,n-p-1}}{1+[\frac{p}{n-p-1}]F_{\frac{\alpha}{2},p,n-p-1}} \\ LCL &= \frac{(m-1)^2}{m} \frac{[\frac{p}{n-p-1}]F_{1-\frac{\alpha}{2},p,n-p-1}}{1+[\frac{p}{n-p-1}]F_{1-\frac{\alpha}{2},p,n-p-1}} \end{aligned}$$

where F_{α,ν_1,ν_2} is the value of the inverse of the F cumulative distribution with ν_1 and ν_2 degrees of freedom, evaluated at the confidence level $(1-\alpha)$.

In Phase II, any outliers identified during Phase I are removed and the remaining observations are used to recalculate the T^2 statistic. In other words, the Phase II T_i^2 statistic is given by $T_i^2 = (\tilde{\boldsymbol{x}}_i - \bar{\boldsymbol{x}})S^{-1}(\tilde{\boldsymbol{x}}_i - \bar{\boldsymbol{x}})^T$, where $\tilde{X} = [\tilde{\boldsymbol{x}}_1, \tilde{\boldsymbol{x}}_2, \ldots, \tilde{\boldsymbol{x}}_n]^T$ is the new observed data matrix, also referred to as the historical data. Again, any historical data that is plotted outside the control limits and had an assignable cause determined are discarded. Phase II verifies if the process is generating and maintaining values that are considered in control. The control limits for Phase II are

$$\begin{aligned} UCL &= \frac{p(n+1)(n-1)}{n(n-p)} F_{\frac{\alpha}{2},p,n-p} \\ LCL &= \frac{p(n+1)(n-1)}{n(n-p)} F_{1-\frac{\alpha}{2},p,n-p} \end{aligned}$$

Unlike the univariate control charts, the T^2 statistic does not represent the original variables and therefore when an out of control situation occurs we can not determine if it was due to an excess variation of a particular variable or due to a change in the covariance/correlation matrix.

To circumvent these problems, the principal component chart may be used. This control chart can detect changes in the covariance/correlation structure and it may indicate the specific domain that created this excess variation [5]. It also has the advantage of reducing the number of dimensions that need to be analyzed [14].

1.2.2 Principal Component Analysis

Principal Component Analysis (PCA) is a method for transforming the observations in a dataset into new observations which are uncorrelated with each other and account for decreasing proportions of the total variance of the original variables. Each new observation is a linear combination of the original observations.

Standardizing the data is often preferable when the variables are in different units or when the variance of the different columns of the data is substantial. The standardized data matrix is given by

$$Z = (X - \mathbf{1}\,\bar{x})D^{-1/2},$$

where $\mathbf{1} = (1, \ldots, 1)^T$ is a $n \times 1$ vector of all 1's, and $D = \text{diag}(S)$ is the diagonal of the covariance matrix.

It is worth pointing out the the covariance matrix R of the standardized data Z is exactly the correlation matrix of the original data, and it is given by $R = D^{-1/2}SD^{-1/2}$.

PCA is then performed by applying eigen-decomposition to the matrix R, that is $R = A\Lambda A^T$ where $A = (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_p)$ is a $p \times p$ matrix of eigenvectors (also called principal components) and $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_p)$ is a diagonal matrix of eigenvalues. These eigenvalues are equal to the variance explained by each of the principal components, in decreasing order of importance. The principal component score matrix is an $n \times p$ data matrix Y given by $Y = ZA = (\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n)^T$ which is the data mapped into the new coordinate system defined by the principal components. Moreover,

$$\text{cov}(Y) = \frac{1}{n-1}Y^TY = \frac{1}{n-1}A^TZ^TZA = \Lambda.$$

Hence, besides retaining the maximum amount of variance in the projected data, PCA also has the following property: the projected data $\boldsymbol{y}_k$ are uncorrelated with variance equal to $\text{var}(\boldsymbol{y}_k) = \lambda_k$ for $k = 1, \ldots, p$.

Assuming we want 99.7% confidence intervals, the upper control limit (UCL), the center line (CL) and the lower control limit (LCL) are given by

$$UCL = +3\sqrt{\lambda_k}$$

$$CL = 0$$

$$LCL = -3\sqrt{\lambda_k}$$

1.2.3 Dimensionality Reduction Techniques

As mentioned earlier, selecting the correct number of eigenvectors k will separate the signals of the process from the random noise. Several techniques exist for determining the value of the reduction order, but there is no apparent dominant technique [3].

The explained variance test determines the number of k components by calculating the number of eigenvectors needed to represent a certain percentage of the total variance. Given that each eigenvalue (λ_i) represents a certain percentage of the variance, the explained variance test equation can be given by

$$\frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{p} \lambda_i} = \alpha\%$$

Since the minimum percentage ($\alpha\%$) is arbitrarily chosen, the number k components needed for each application may be too low or too high [2, 3].

The scree plot shown in Fig. 1.3, plots the eigenvalues in decreasing magnitude. The number of components k is chosen at the knee where the graph changes from being vertical to becoming horizontal. In Fig. 1.3 we can observe a typical scree plot for a data matrix containing 7 variables. In this case, we would choose 2 components. The identification of the number of components may be ambiguous to identify and therefore hard to automate [3].

The pareto chart is a combination of the two previous methods discussed. It linearly plots the accumulative percentage of each eigenvalue and display the individual percent variance contribution of each component in decreasing order in a bar graph. We can select the k components by observing the break of the linear curve. If the break is between two components, we may look at the bar graph and decide on the total variance each component will additional contribute. By looking at the pareto chart in Fig. 1.4 we can observe that $k = 2$ seems to be the appropriate choice. In order to simplify the automation, we selected all the components whose cumulative explained variance is greater or equal to 95 percent as our cutoff (approximately 2σ).

1.2.4 Electroencephalograms

An EEG is a very useful, non-invasive way, of monitoring brain activity. Electrodes are placed on specific locations on the scalp and recordings are made of the signals that are detected. The trained EEG interpreter

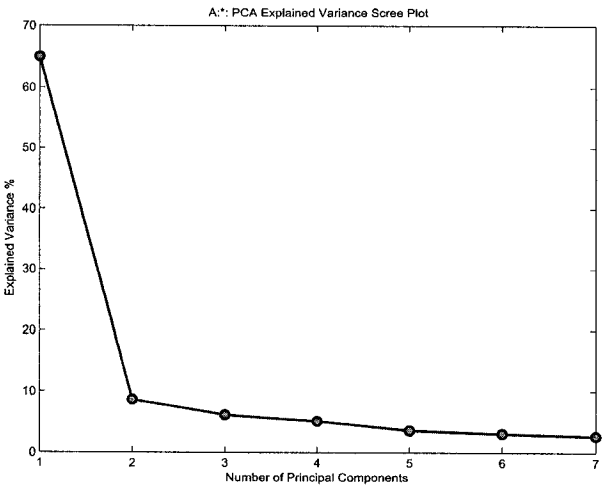

Figure 1.3: Scree plot.

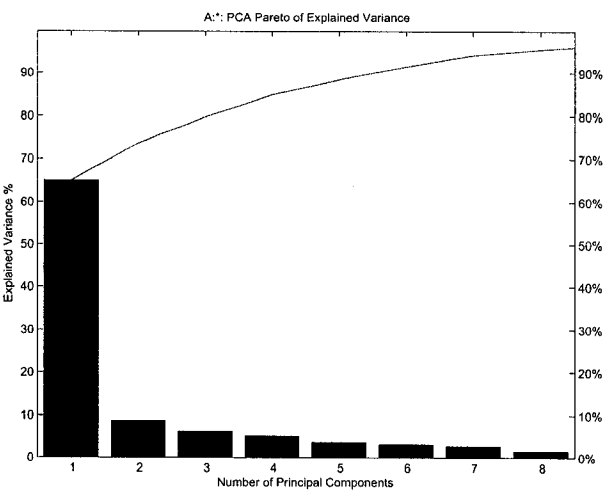

Figure 1.4: Pareto chart.

can determine whether a person is asleep, awake, or is having some sort of a seizure, simply by looking at the data that is recorded [13]. EEG reading involves the interpretation of wave forms largely by their frequency and to a lesser extent by the morphology of the wave or of the wave complex of several waves. The difficulty lies, in part, in recognizing artifacts and also in being able to differentiate normal variants from abnormalities. Frequency means the number of waves per unit time (i.e., per second). The frequencies of the EEG waves run from 0.5 per second to hundreds/second. Waves are usually defined by their frequency and are divided, on this basis, into: alpha, beta, theta, and delta. Certain waves have characteristic forms irrespective of their frequency and are recognizable by their shape. In other instances, pairs or groups of waves have typical appearances. Single waves that are specially shaped include, for instance spikes or sharp waves - waves that rise rapidly to a point and fall away equally dramatically with a base that is small compared to the wave's amplitude. Some wave forms can be recognized by their morphology and these include two main types: specially shaped waves, and specially shaped wave complexes [28].

A scalp-recorded EEG is generated by the pooled activity of billions of cortical neurons influenced by shared activity between cortical and subcortical regions [4]. Each EEG electrode records activity from multiple generators of EEG activity. A standard for the placement of electrodes was established [10]. This standard was known as the 10-20 International System of Electrode Placement. The letters F, C, T, P, and O refer to the "frontal", "central", "temporal", "parietal", and "occipital" regions of the skull. The odd numbers refer to left hemisphere sites and even numbers refer to right hemisphere sites. Thus, C4 refers to the right central region. The term "10-20" refers to the placement of electrodes placed 10 percent or 20 percent of the total distance between specified skull locations. Studies have shown that these placements correlate with the corresponding cerebral cortical regions [10]. Of the 21 electrodes used, 19 are used for scalp sites recording cortical areas and 2 are placed on the earlobes as reference electrodes. Figure 1.5 depicts the regions on the scalp [11].

Two basic types of EEG montages are used: Referential (mono-polar), and Bipolar. The referential montage involves collecting information at a specific electrode relative to a common reference electrode. The main advantage of referential montages is that the common reference allows valid comparison of activity in many different electrode pairings. Mono-polar montages have the disadvantage where no single reference is ideal. Reference sites may pick up EEG activity. Another disadvantage is that electro-myographic (EMG)

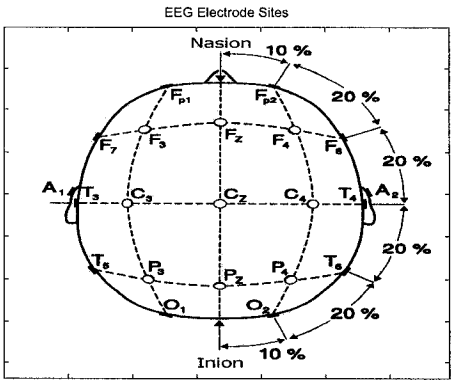

Figure 1.5: EEG Measurement Sites on the Scalp.

or heart-beat (EKG) artifacts may occur [10].

Bipolar montages show the difference in activity between two active scalp sites. A major advantage of bipolar montages is that localization of electrophysiological events is easier. By examining a sequence of bipolar derivations, the occurrence of a specific spike of the EEG can be found when searching for a phase reversal of the electrical signal as one moves spatially from left to right [10].

The EEG data we analyzed contained of the following channels:

1. Fp1-F7 is the Left Frontal Pole region to Left Frontal region
2. F7-T3 is the Left Frontal region to Left Temporal region
3. T3-T5 is the Left Temporal region
4. T5-O1 is the Left Temporal region to Left Occipital region
5. Fp2-F8 is the Right Frontal Pole region to Right Frontal region
6. F8-T4 is the Right Frontal region to Right Temporal region
7. T4-T6 is the Right Temporal region
8. T6-O2 is the Right Temporal region to Right Occipital region

9. T3-Cz is the Left Temporal region to Central Vertex region
10. Cz-T4 is the Scalp Vertex region to Left Temporal region
11. Fp1-F3 is the Left Frontal Pole region
12. F3-C3 is the Left Frontal region to Left Central region
13. C3-P3 is the Left Central region to Left Parietal region
14. P3-O1 is the Left Parietal region to Left Occipital region
15. Fp2-F4 is the Right Frontal Pole region to Right Frontal region
16. F4-C4 is the Right Frontal region to Right Central region
17. C4-P4 is the Right Central region to Right Parietal region
18. P4-O2 is the Right Parietal region to Right Occipital region
19. O1 is the Left Occipital region
20. O2 is the Right Occipital region

1.3 Contributions

The contributions of this thesis are as follows:

☞ **Kernel Principal Component Analysis of EEG Data:** We present a user-friendly graphical interface to analyze EEG data using multivariate statistical dimensionality-reduction techniques. The use of multivariate statistical process control is facilitated by the proliferation of sensor data that is typically complex, high-dimensional and generally correlated. Electroencephalographs (EEGs) consist of vast amounts of complex data that require a trained professional to perform a proper analysis. We apply a robust multivariate statistical process control chart using kernel principal component analysis to the analysis of EEGs. The proposed control chart is effective in the detection of outliers, and its control limits are derived from the eigen-analysis of the Gaussian kernel matrix in the Hilbert feature space.

Our experimental results demonstrate the effectiveness of dimensionality reduction using PCA and kernel PCA in analyzing multivariate data.

☞ **Kernel Isometric Mapping of EEG Data:** We introduce a multivariate statistical process control chart using kernel isometric feature maps (kernel Isomap). The first step is to creates an kernel Isomap matrix and then perform eigen-analysis on the mapped data in the second step. The proposed control chart is effective in the detection of outliers, and its control limits are derived from the eigen-analysis of the kernel matrix in the Hilbert feature space. Our experimental results demonstrate the effectiveness of dimensionality reduction using Kernel Isomap control charts in analyzing multivariate data.

☞ **Kernel Locally Linear Embedding Algorithm for Outliers Detection:** We introduce a new multivariate statistical process control chart for outliers detection using kernel local linear embedding algorithm. The proposed control chart is effective in the detection of outliers, and its control limits are derived from the eigen-analysis of the kernel matrix in the Hilbert feature space. Our experimental results show the much improved performance of the proposed control chart in comparison with existing multivariate monitoring and controlling charts.

1.4 Thesis Overview

The organization of this thesis is as follows:

- ❑ The first Chapter contains a brief review of essential concepts and definitions which we will refer to throughout the thesis, and presents a short summary of material relevant to multivariate statistical process control.

- ❑ In Chapter 2, we introduce principal component analysis and kernel principal component analysis as effective methods for dimensionality reduction of multivariate data. The effectiveness of our method will be demonstrated through the analysis of EEGs.

- ❑ In Chapter 3, we introduce kernel Isomap as a new multivariate statistical process control chart for dimensionality reduction. A detailed description of the algorithm and EEG analysis results will be presented to demonstrate the performance of the proposed approach.

- In Chapter 4, we introduce kernel LLE as a new multivariate statistical process control chart for dimensionality reduction. A detailed description of the algorithm and comparison with similar methodologies will be presented.

- In the **Conclusions** Chapter, we summarize the various methodologies that were presented for dimensionality reduction, and we propose several future research directions that are directly or indirectly related to the work performed in this thesis.

CHAPTER 2

Kernel Principal Component Analysis of EEG Data

The use of multivariate statistical process control is facilitated by the proliferation of sensor data that is typically complex, high-dimensional and generally correlated. Almost every new experiment or application requires the availability of considerable amount of data for the purpose of analysis. Depending on the type of data, various techniques of data analysis are used to extract useful information and facilitate conclusions.

Electroencephalographs (EEGs) are one such example whereby vast amounts of complex data are present and require a trained professional many hours to perform a proper analysis. In this chapter, we present a multivariate statistical dimensionality-reduction technique for analyzing EEGs. Our experimental results demonstrate the effectiveness of our proposed approach in the analysis of multivariate data [12].

2.1 Introduction

PCA is an exploratory multivariate data reduction technique for simplifying complex data. In a data set with many correlations, we need a technique to look at the overall structure of the data. PCA is based on linear correlation and it transforms original variables into new uncorrelated variables. The goal of PCA is to reduce the dimensionality of a data set (sample) by finding a new set of variables, smaller than the original set of variables which retains most of the sample's information. Moreover, PCA can be used to reduce number of dimensions in data, find patterns in high-dimensional data, or visualize data of high dimensionality.

In this chapter, we present a multivariate statistical quality control methodology for analyzing EEG data. The primary motivation behind using multivariate statistical techniques is to reduce the data so that it can be interpreted by a trained person in minutes rather than hours. Multivariate statistical techniques have been

used extensively to analyze the data in various industries over the last three decades. Recently, sensor technology is making gigabytes upon gigabytes of data available for analysis and this collected data is most often multivariate in nature [1] [30]. In this chapter, we demonstrate the effectiveness of multivariate statistical techniques and in particular principal component analysis (PCA) and kernel PCA [24] in analyzing and interpreting EEG data.

Kernel Principal Component Analysis (KPCA) provides a set of nonlinear axes in the input space, thus allowing it to represent complex data distributions with a small number of axes. We apply a multivariate statistical process control chart using kernel principal component analysis to the analysis of EEGs. The proposed control chart is effective in the detection of outliers, and its control limits are derived from the eigenanalysis of the Gaussian kernel matrix in the Hilbert feature space. Our experimental results demonstrate the ability to detect events in EEGs.

2.2 Problem Formulation

PCA is a powerful technique for data reduction and visualization. It involves a mathematical procedure that transforms a number of possibly correlated variables into a smaller number of uncorrelated variables, called Principal Components (PCs). PCA is based on linear correlation and it transforms original variables into new uncorrelated variables that represent the original data using fewer dimensions. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible. The first principal component goes through the centroid and minimizes the square of the distance of each point to that line. The second principal component also goes through the centroid, and also goes through the maximum variation in the data, but is orthogonal to the first Let X represent the EEG data, with m observations and p channels: principal component in order to be uncorrelated.

$$X_{m,p} = \begin{pmatrix} x_{11} \dots & x_{1p} \\ \vdots & \vdots \\ x_{m1} \dots & x_{mp} \end{pmatrix}.$$

The mean and the covariance matrix of the k retained components of the EEG data are given by:

$$\bar{x} = \frac{1}{m}\sum_{i=1}^{m} x_{ij} \quad \text{and} \quad S_j^2 = \frac{1}{m-1}\sum_{i=1}^{m}(x_{ij} - \bar{x})^2, i = 1, 2, ..., m, j = 1, 2, .., k$$

are the sample mean and covariance matrix respectively.

Since the various channels of the EEG have potentially different units of measurement and/or magnitudes, we standardize the data:

$$Z_{ij} = \frac{x_{ij} - \bar{x}_j}{s_j}, i = 1, 2, , m, j = 1, 2, , k$$

The j^{th} principal component is a linear combination of the standardized variables:

$$Y_j = c_{1j}Z_1 + c_{2j}Z_2 + \ldots + c_{kj}Z_k, j = 1, 2, , k$$

The principal component scores of the i^{th} observation for the j^{th} principal component are as follows:

$$y_{ij} = a_{1j}z_{i1} + a_{2j}z_{i2} + \ldots + a_{kj}z_{ik}, j = 1, 2, , k$$

The variances of the k principal component scores are given by the eigenvalues of the covariance matrix:

$$\lambda_1, \lambda_2, \ldots \lambda_k.$$

2.3 Proposed Method

PCA is a linear transformation of the original data set into a smaller number of components while keeping most of the variance [14]. PCA has two main advantages over other multivariate statistical process control techniques. First, the principal components are uncorrelated and second, only a few components are needed in order to capture most of the variance [5].

Kernel principal component analysis is a nonlinear generalization of PCA, and consists of mapping the data to a higher (possibly infinite) dimensional feature space via a nonlinear map, and then computing the dot products in the feature space. Suppose we have an input data set $X = \{\boldsymbol{x}_i : i = 1, \ldots, m\}$ where each observation $\boldsymbol{x}_i$ is an p-dimensional vector and the distribution of the data is nonlinear. The Kernel PCA algorithm consists of two main steps: the first step is to linearize the distribution of the input data

by using a nonlinear mapping $\Phi : \mathbb{R}^p \rightarrow \mathcal{F}$ from the input space $\mathbb{R}^p$ to a higher-dimensional (possibly infinite-dimensional) feature space $\mathcal{F}$. The mapping Φ is defined implicitly, by specifying the form of the dot product in the feature space. In other words, given any pair of mapped data points, the dot product is defined in terms of a kernel function $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Phi(\boldsymbol{x}_i) \cdot \Phi(\boldsymbol{x}_j)$.

The most commonly used kernel is the Gaussian kernel $K(\boldsymbol{x}, \boldsymbol{y}) = \exp(-\|\boldsymbol{x} - \boldsymbol{y}\|^2/(2\sigma^2))$ with parameter σ. In the second step, PCA in applied to the mapped data set $\Phi = \{\Phi_i : i = 1, \ldots, m\}$ in the feature space, where $\Phi_i = \Phi(\boldsymbol{x}_i)$. The second step of kernel PCA is to apply PCA in the feature space by performing an eigendecomposition on the covariance matrix of the mapped data which is given by

$$C = \frac{1}{m-1} \sum_{i=1}^{m} \widetilde{\Phi}(\boldsymbol{x}_i)^T \widetilde{\Phi}(\boldsymbol{x}_i)$$

where $\widetilde{\Phi}(\boldsymbol{x}_i) = \Phi(\boldsymbol{x}_i) - (1/m) \sum_{i=1}^{m} \Phi(\boldsymbol{x}_i)$ is the centered mapped data.

The eigenvectors of C are given by

$$\boldsymbol{v} = \frac{1}{\mu} C\boldsymbol{v} = \sum_{i=1}^{m} \widetilde{\Phi}(\boldsymbol{x}_i) \left(\frac{1}{\mu(m-1)} \widetilde{\Phi}(\boldsymbol{x}_i)^T \boldsymbol{v} \right) = \sum_{i=1}^{m} \alpha_i \widetilde{\Phi}(\boldsymbol{x}_i),$$

where $\alpha_i = (\widetilde{\Phi}(\boldsymbol{x}_i)^T \boldsymbol{v})/(\mu(m-1))$. In other words, an eigenvector of C is a linear combination of $\{\widetilde{\Phi}(\boldsymbol{x}_i)\}$. Taking the dot product of $\widetilde{\Phi}(\boldsymbol{x}_j)$ with $\boldsymbol{v}$ yields

$$\widetilde{\Phi}(\boldsymbol{x}_j) \cdot \boldsymbol{v} = \sum_{i=1}^{m} \alpha_i \widetilde{\Phi}(\boldsymbol{x}_i) \cdot \widetilde{\Phi}(\boldsymbol{x}_j) = \sum_{i=1}^{m} \alpha_i \widetilde{K}_{ij},$$

which implies that $\mu(m-1)\alpha_j = \sum_{i=1}^{m} \alpha_i \widetilde{K}_{ij}$. Hence

$$\widetilde{K}\boldsymbol{\alpha} = \tilde{\mu}\boldsymbol{\alpha},$$

where $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_m)$ and $\tilde{\mu} = \mu(m-1)$. That is, $\boldsymbol{\alpha}$ is an eigenvector of $\widetilde{K}$. If the eigenvectors of C are orthonormal (i.e. $\boldsymbol{v}^T\boldsymbol{v} = 1$) then

$$\begin{aligned} 1 = \boldsymbol{v}^T\boldsymbol{v} &= \sum_{i,j=1}^{m} \alpha_i \alpha_j \widetilde{\Phi}(\boldsymbol{x}_i) \cdot \widetilde{\Phi}(\boldsymbol{x}_j) = \sum_{i,j=1}^{m} \alpha_i \alpha_j \widetilde{K}_{ij} \\ &= \boldsymbol{\alpha}^T \widetilde{K} \boldsymbol{\alpha} = \mu(m-1)\boldsymbol{\alpha}^T\boldsymbol{\alpha} \end{aligned}$$

and hence $\|\boldsymbol{\alpha}\| = 1/\sqrt{\mu(m-1)}$.

Algorithm: Kernel Principal Component Control Chart

1. Choose the appropriate σ for the Gaussian kernel matrix
2. Construct the kernel matrix $K = (K_{ij})$ of the mapped data: $K_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Phi(\boldsymbol{x}_i) \cdot \Phi(\boldsymbol{x}_j)$.
3. Construct the kernel matrix $\widetilde{K} = HKH$ of the centered mapped data, where $H = I - J/n$ the centering matrix is defined in terms of the identity matrix I and the matrix of all ones J.
4. Find the largest p eigenvectors $\boldsymbol{\alpha}_r$ $(r = 1, \ldots, p)$ of $\widetilde{K}$ and their corresponding eigenvalues $\tilde{\mu}_r$.
5. Given a test point $\boldsymbol{x}$ with image $\Phi(\boldsymbol{x})$, compute the projections onto the eigenvectors $\boldsymbol{v}_r$ given by the equation

$$\boldsymbol{v}_r \cdot \widetilde{\Phi}(\boldsymbol{x}) = \frac{1}{\sqrt{(n-1)}} \sum_{i=1}^{m} \alpha_i \widetilde{\Phi}(\boldsymbol{x}_i) \cdot \widetilde{\Phi}(\boldsymbol{x})$$

Table 2.1: Algorithmic steps for Kernel PCA.

The main algorithmic step of the proposed kernel principal component chart as shown in Table 2.1.

Assuming we want $\pm 3\sigma$ confidence intervals, the upper control limit (UCL), the center line (CL), and the lower control limit (LCL) of the kernel principal component chart are

$$UCL = +3\sqrt{\mu_r}$$

$$CL = 0$$

$$LCL = -3\sqrt{\mu_r}$$

2.4 Experimental Results - PCA

In this section, we demonstrate the effectiveness of PCA, as implemented in the GUI we developed. The analysis was repeated on three hours of data of three patient files containing various events. It can be seen that PCA control charts detected outliers. The results are summarized in the following subsections.

Analysis of Patient File 'A'

Fig. 2.1 through Fig. 2.11 show the charts generated by analyzing an EEG from a sample patient file. The first four charts represent the first four Eigenvectors related to the first four principal components. Figure 2.6 depicts a pareto diagram of the variance of each principal component relative to the sum of all principal components. Figure 2.7 depicts a scree plot of the variance due to each principal component. Typically, the knee of the scree plot is used to determine how many principal components will be used to represent the data. In our case, we used 95 percent as our cutoff. That is to say, 95 percent of the variance of the first 1000 seconds of the data in patient file 'A' can be attributed to seven principal components.

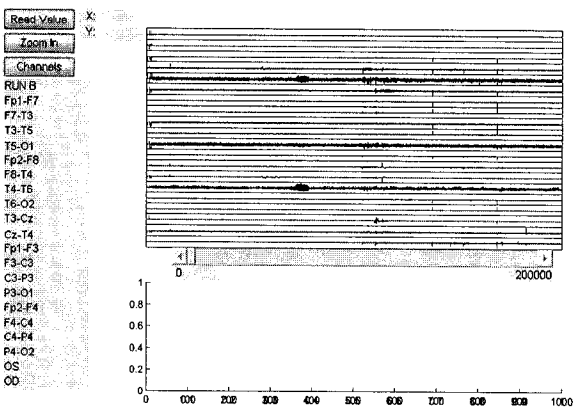

Figure 2.1: User Interface: EEG Channels Read into Software Tool.

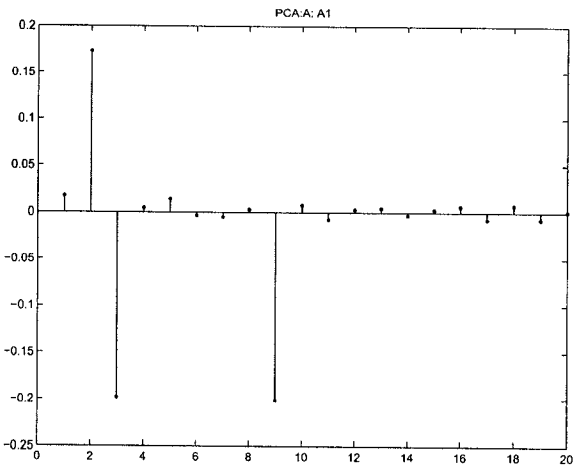

Figure 2.2: PCA: Eigenvector A1 From Sample 0 to Sample 200000 of Patient 'A' File.

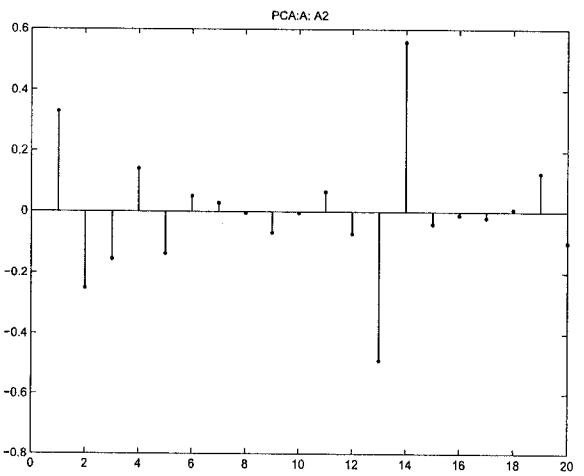

Figure 2.3: PCA: Eigenvector A2 From Sample 0 to Sample 200000 of Patient 'A' File.

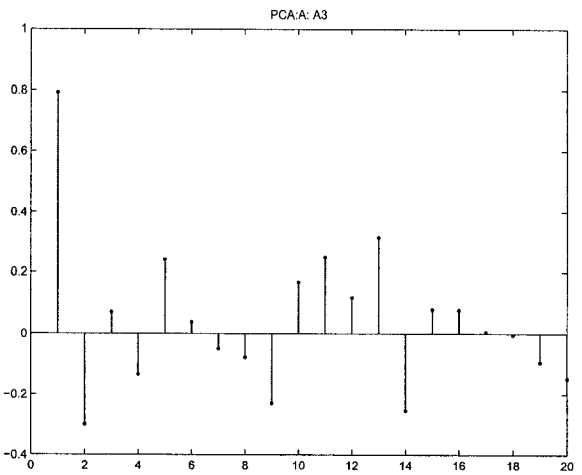

Figure 2.4: PCA: Eigenvector A3 From Sample 0 to Sample 200000 of Patient 'A' File.

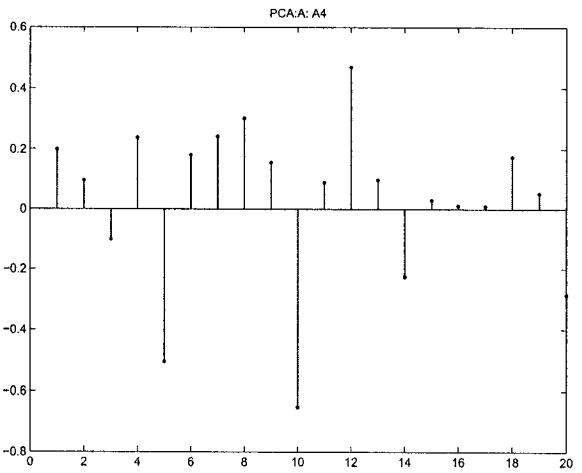

Figure 2.5: PCA: Eigenvector A4 From Sample 0 to Sample 200000 of Patient 'A' File.

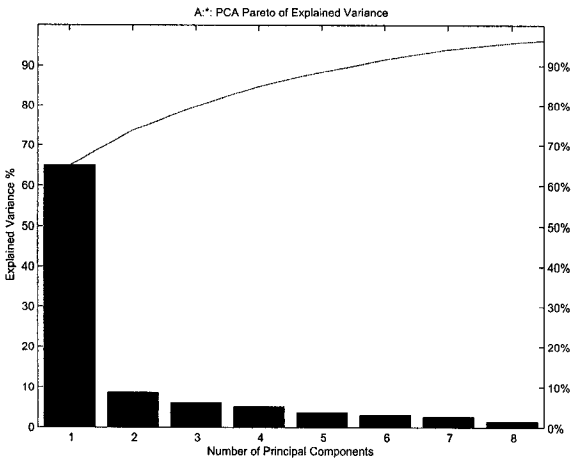

Figure 2.6: PCA: Pareto of Explained Variance From Sample 0 to Sample 200000 of Patient 'A' File.

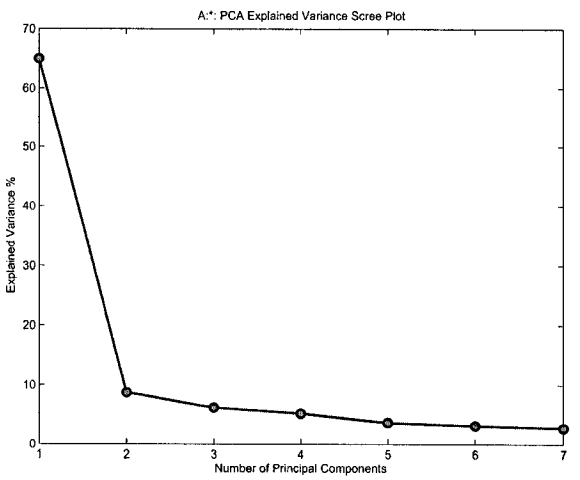

Figure 2.7: PCA: Scree Plot of Explained Variance From Sample 0 to Sample 200000 of Patient 'A' File.

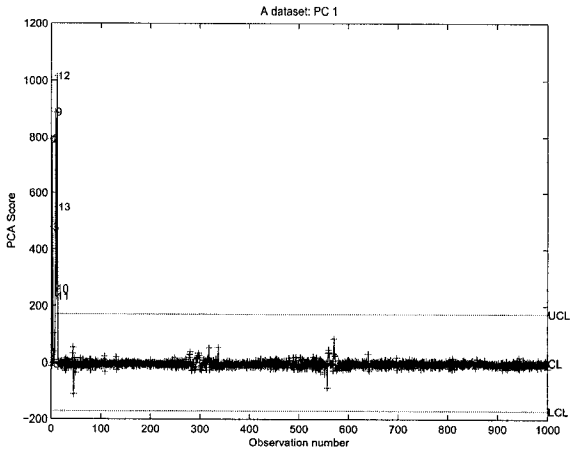

Figure 2.8: PCA Control Chart of First Principal Component From Sample 0 to Sample 200000 of Patient 'A' File.

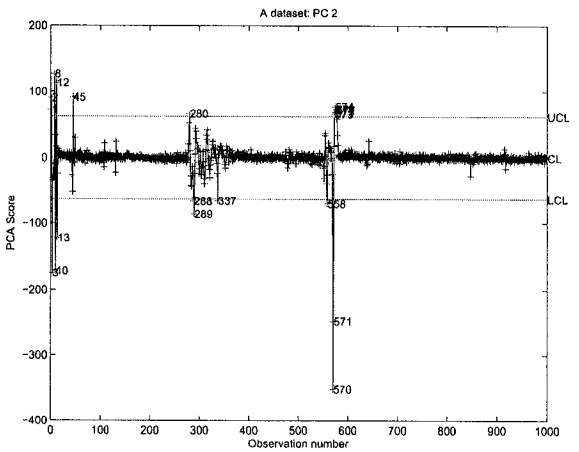

Figure 2.9: PCA Control Chart of Second Principal Component From Sample 0 to Sample 200000 of Patient 'A' File.

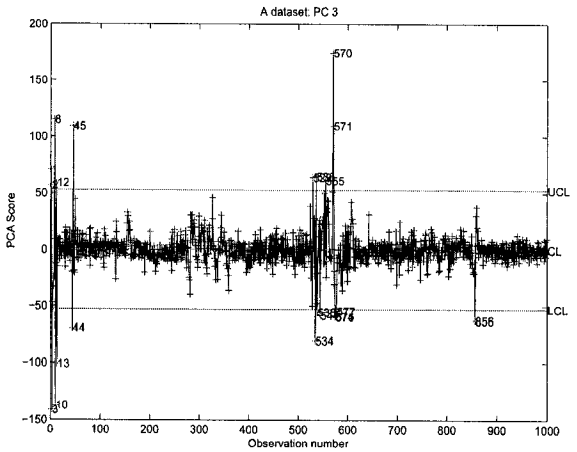

Figure 2.10: PCA Control Chart of Third Principal Component From Sample 0 to Sample 200000 of Patient 'A' File.

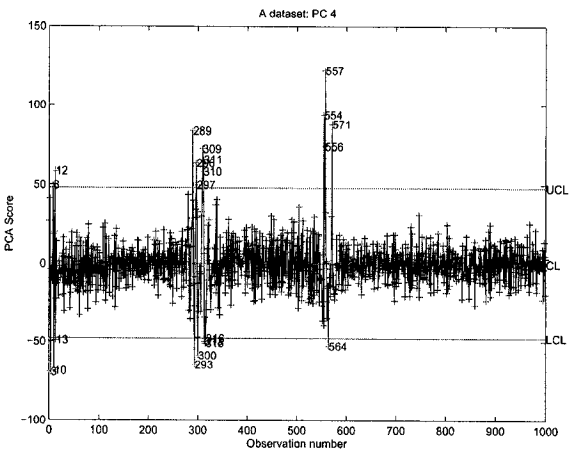

Figure 2.11: PCA Control Chart of Fourth Principal Component From Sample 0 to Sample 200000 of Patient 'A' File.

Analysis of Patient File 'B'

Figure 2.12 through Figure 2.22 were generated by analyzing an EEG from patient file "B". The first four charts represent the Eigenvectors of the first four principal components. Figure 2.17 depicts a pareto diagram of the variance of each principal component relative to the sum of all principal components. Figure 2.18 depicts a scree plot of the variance due to each principal component.

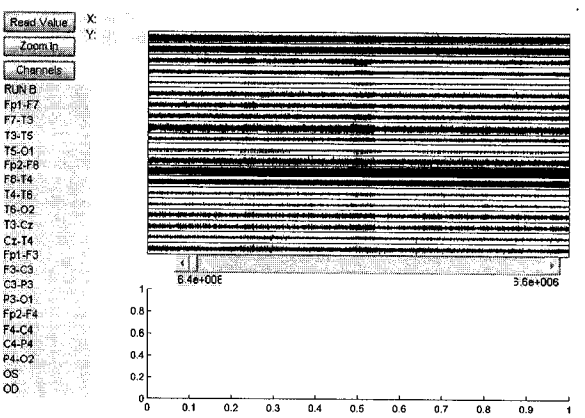

Figure 2.12: User Interface: EEG Channels Read into Software Tool.

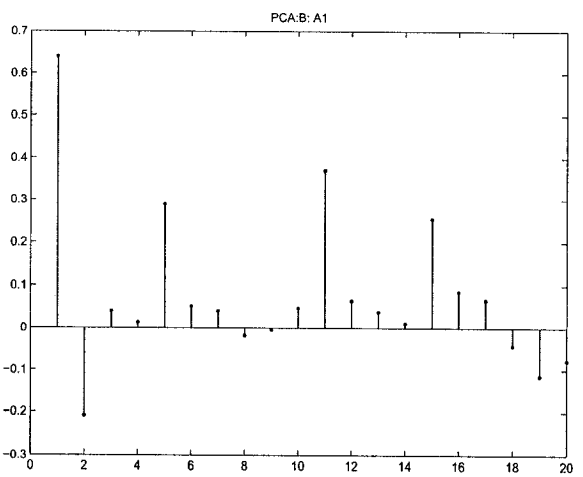

Figure 2.13: PCA: Eigenvector A1 From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

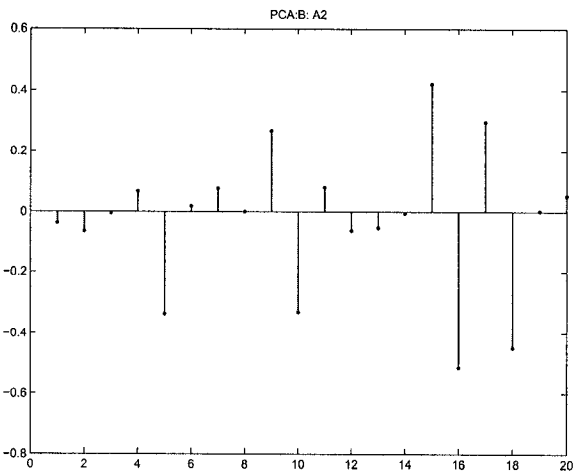

Figure 2.14: PCA: Eigenvector A2 From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

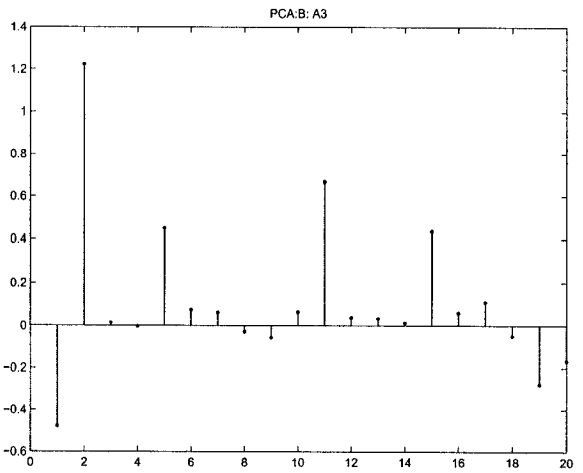

Figure 2.15: PCA: Eigenvector A3 From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

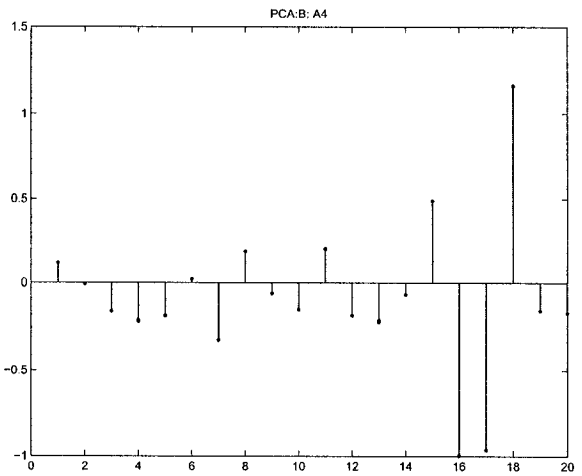

Figure 2.16: PCA: Eigenvector A4 From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

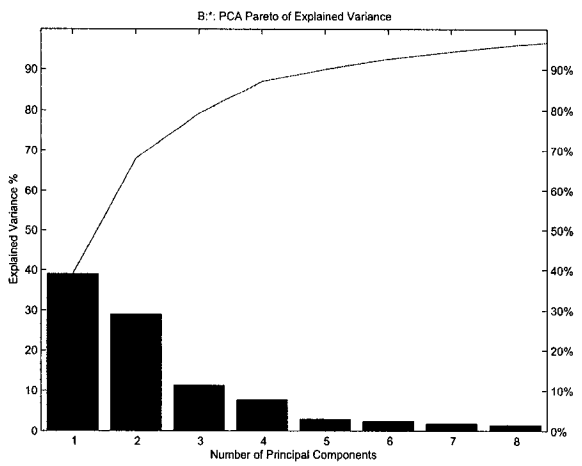

Figure 2.17: PCA: Pareto of Explained Variance From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

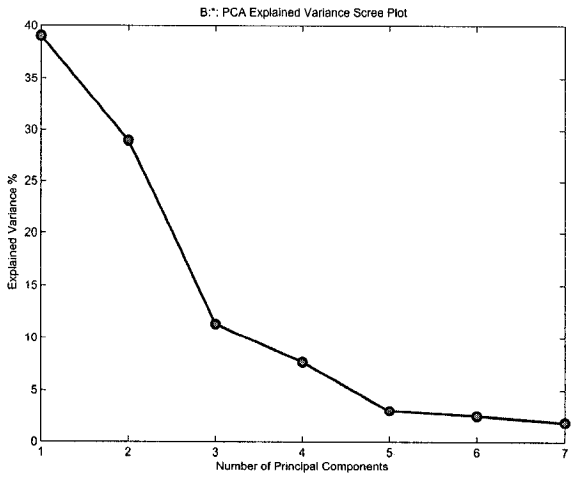

Figure 2.18: PCA: Scree Plot of Explained Variance From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

As can be seen from the charts above, seven principal components represent 95 percent of the variance for this slice of data of patient "B" file. Clearly, seven principal components is a dimensionality reduction relative to twenty input channels. This is also demonstrated by the principal component control charts below (Figure 2.19 through 2.22).

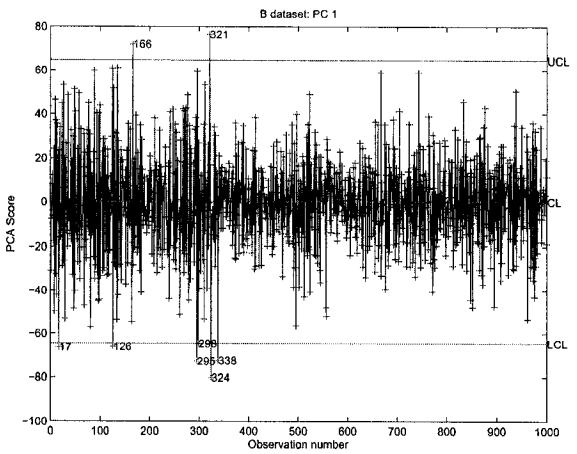

Figure 2.19: PCA Control Chart of First Principal Component From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

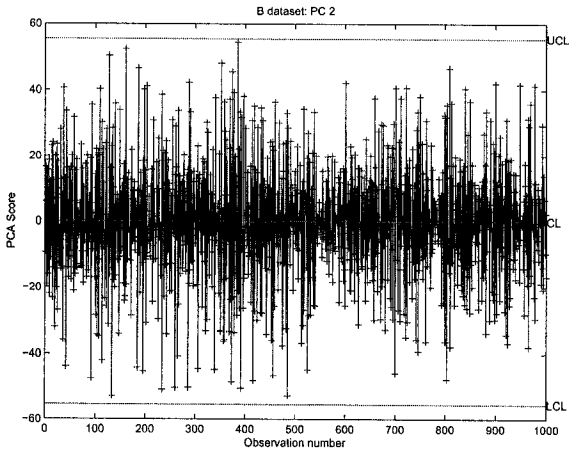

Figure 2.20: PCA Control Chart of Second Principal Component From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

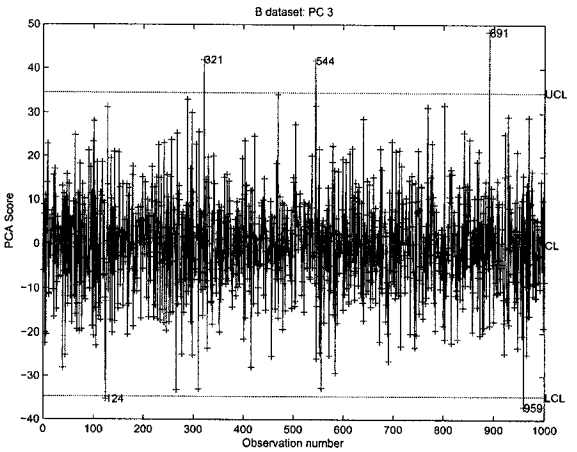

Figure 2.21: PCA Control Chart of Third Principal Component From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

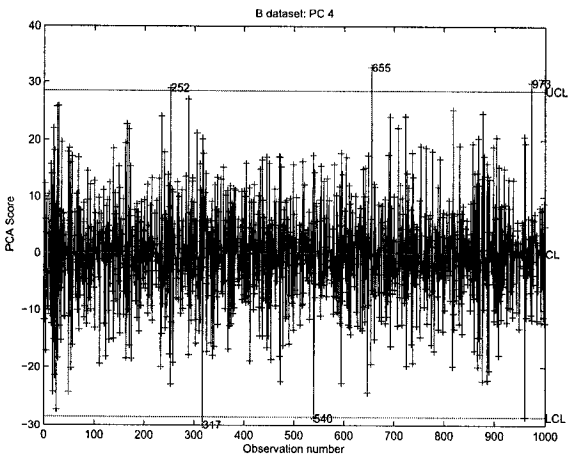

Figure 2.22: PCA Control Chart of Fourth Principal Component From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

Analysis of Patient File 'C'

Figure 2.23 through Figure 2.29 were generated by analyzing an EEG from patient file "C". Figure 2.26 depicts a pareto diagram of the variance of each principal component relative to the sum of all principal components. Figure 2.27 depicts a scree plot of the variance due to each principal component.

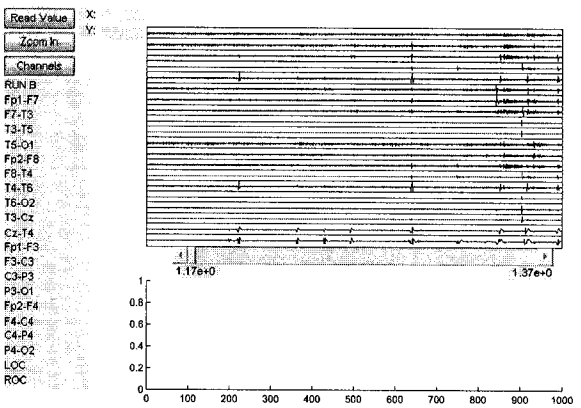

Figure 2.23: User Interface: EEG Channels Read into Software Tool.

Note that only two principal components are required to explain away over 95 percent of the variance for patient file "C" in the interval between sample 1.17e+006 and sample 1.37e+006.

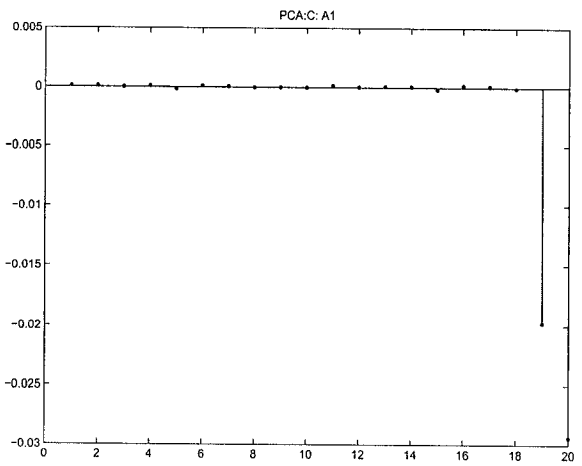

Figure 2.24: PCA: Eigenvector A1 From Sample 1.17e+006 to Sample 1.37e+006 of Patient 'C' File.

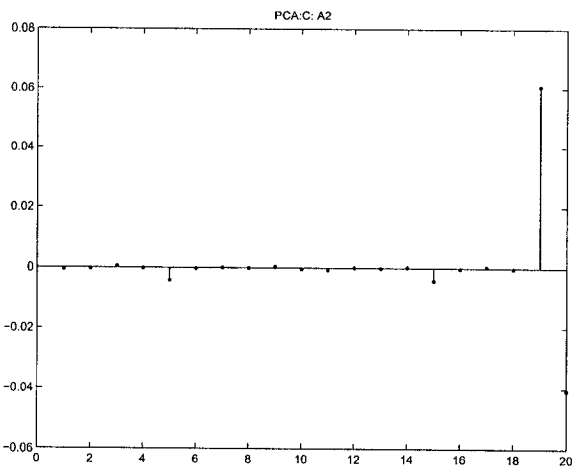

Figure 2.25: PCA: Eigenvector A2 From Sample 1.17e+006 to Sample 1.37e+006 of Patient 'C' File.

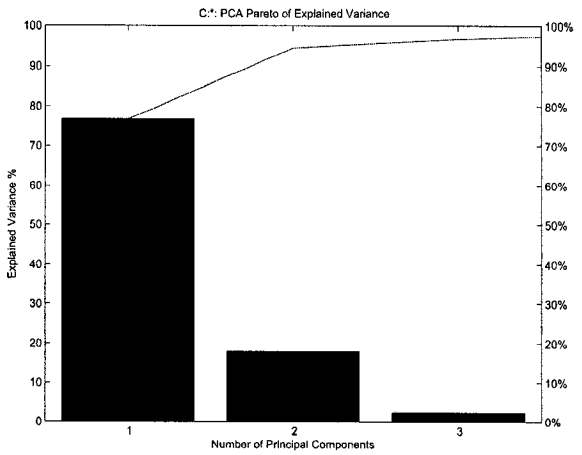

Figure 2.26: PCA: Pareto of Explained Variance From Sample 1.17e+006 to Sample 1.37e+006 of Patient 'C' File.

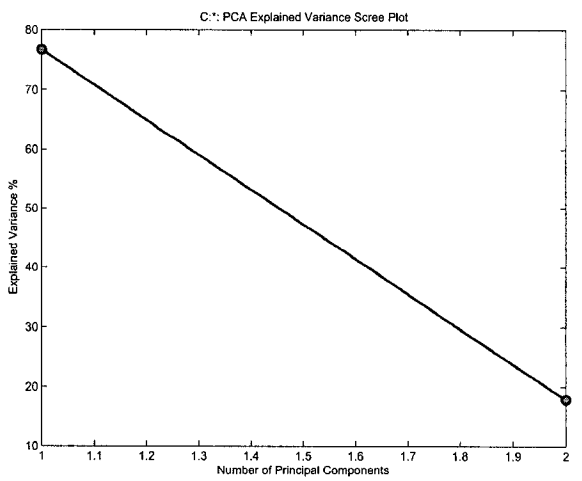

Figure 2.27: PCA: Scree Plot of Explained Variance From Sample 1.17e+006 to Sample 1.37e+006 of Patient 'C' File.

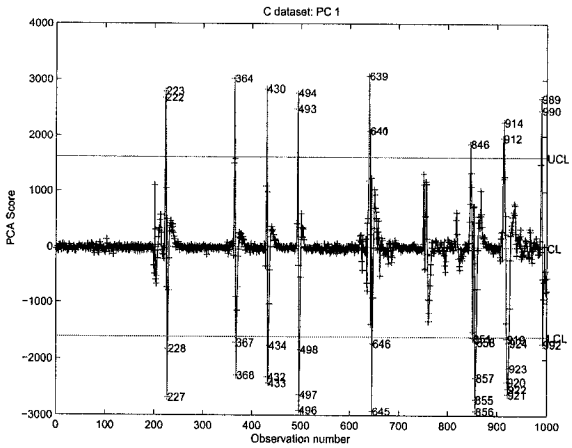

Figure 2.28: PCA Control Chart of First Principal Component From Sample 1.17e+006 to Sample 1.37e+006 of Patient 'C' File.

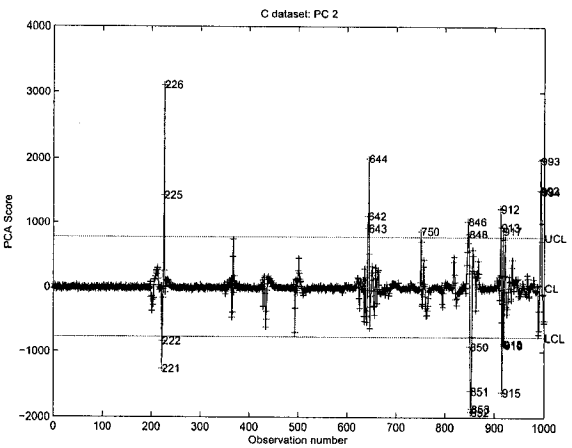

Figure 2.29: PCA Control Chart of Second Principal Component From Sample 1.17e+006 to Sample 1.37e+006 of Patient 'C' File.

2.4.1 Experimental Results - Kernel PCA

In this subsection, we demonstrate the effectiveness of KPCA by repeating the analysis on the same three EEGs and at the same intervals using Kernel PCA. It can be seen that KPCA control charts detected outliers. The results are summarized in the following subsections. In all the experiments, the width of the Gaussian kernel is estimated as follows

$$\sigma = \frac{2}{m(m-1)} \sum_{i<j}^{m} \|z_i - z_j\|,$$

where $[z_1, \ldots, z_m]^T$ is the standardized data.

Analysis of Patient File 'A'

Fig. 2.30 through Fig. 2.40 show the charts generated by analyzing an EEG from a sample patient file. Figure 2.35 depicts a pareto diagram of the variance of each principal component relative to the sum of all principal components. Figure 2.36 depicts a scree plot of the variance due to each principal component.

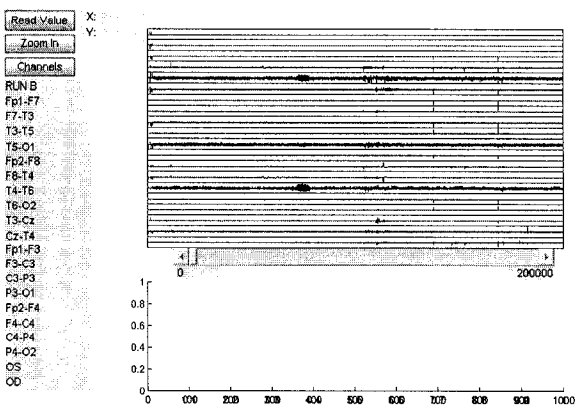

Figure 2.30: User Interface: EEG Channels Read into Software Tool.

As can be seen from the scree plot and pareto diagram, each principal component accounts for very little of the variance. However, useful results are still possible as will be seen when the outliers/events are classified later on. A few of the eigenvectors and control charts are presented, for comparison purposes with PCA and the other techniques described in later chapters.

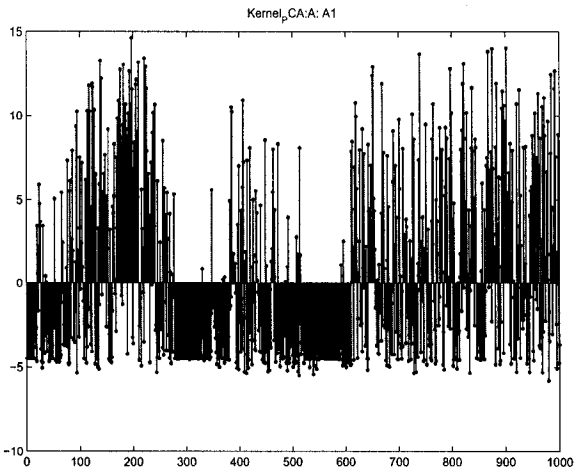

Figure 2.31: Kernel PCA: Eigenvector A1 From Sample 0 to Sample 200000 of Patient 'A' File.

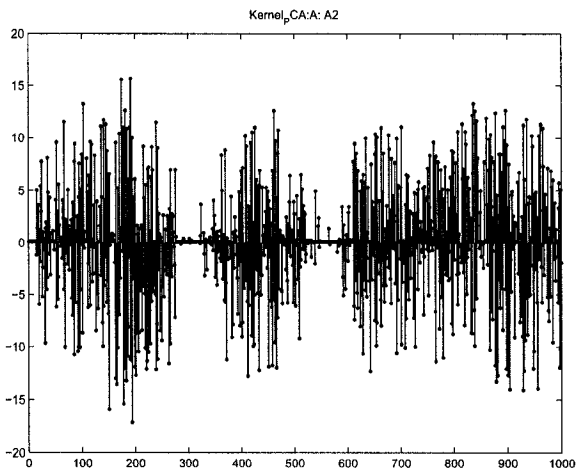

Figure 2.32: Kernel PCA: Eigenvector A2 From Sample 0 to Sample 200000 of Patient 'A' File.

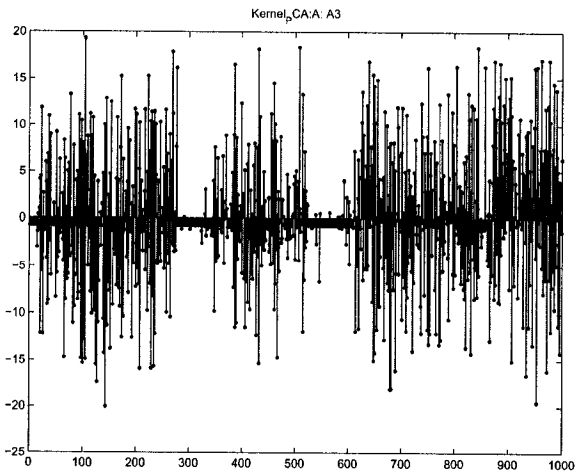

Figure 2.33: Kernel PCA: Eigenvector A3 From Sample 0 to Sample 200000 of Patient 'A' File.

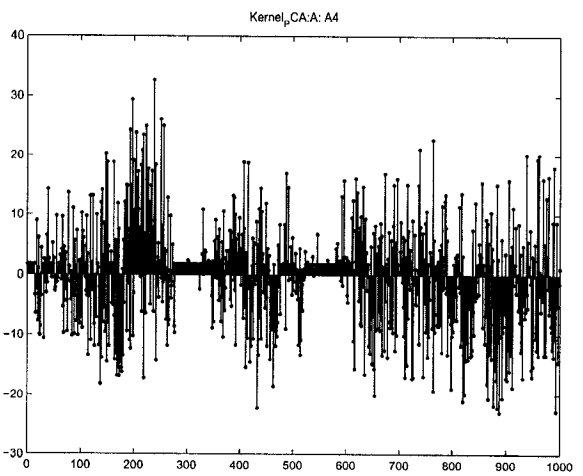

Figure 2.34: Kernel PCA: Eigenvector A4 From Sample 0 to Sample 200000 of Patient 'A' File.

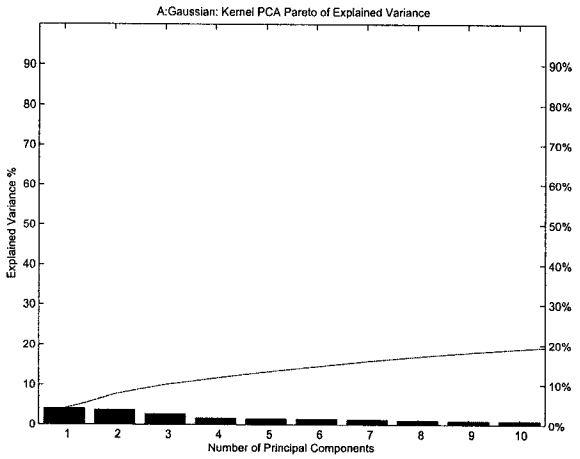

Figure 2.35: Kernel PCA: Pareto of Explained Variance From Sample 0 to Sample 200000 of Patient 'A' File.

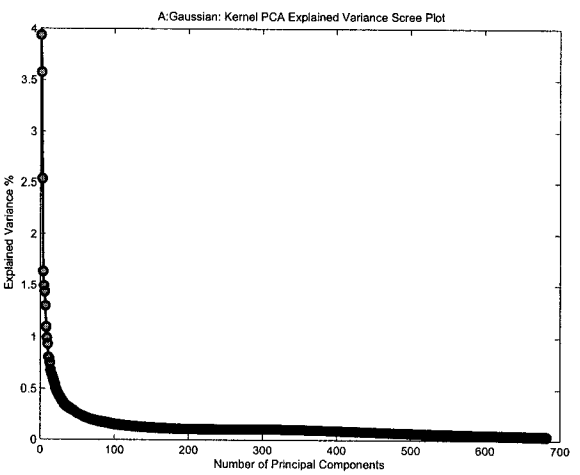

Figure 2.36: Kernel PCA: Scree Plot of Explained Variance From Sample 0 to Sample 200000 of Patient 'A' File.

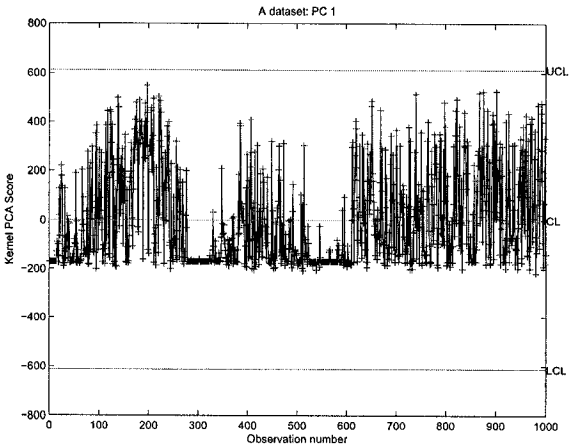

Figure 2.37: Kernel PCA Control Chart of First Principal Component From Sample 0 to Sample 200000 of Patient 'A' File.

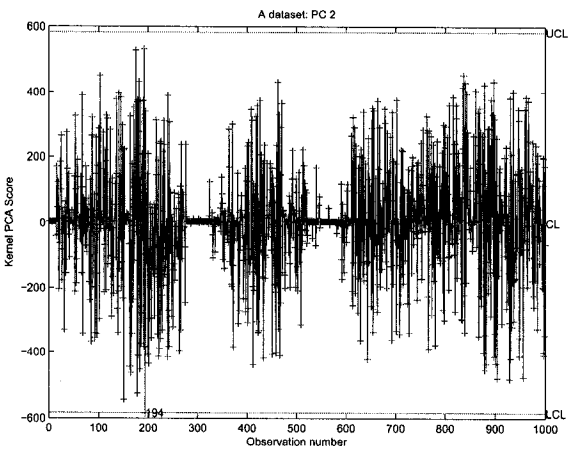

Figure 2.38: Kernel PCA Control Chart of Second Principal Component From Sample 0 to Sample 200000 of Patient 'A' File.

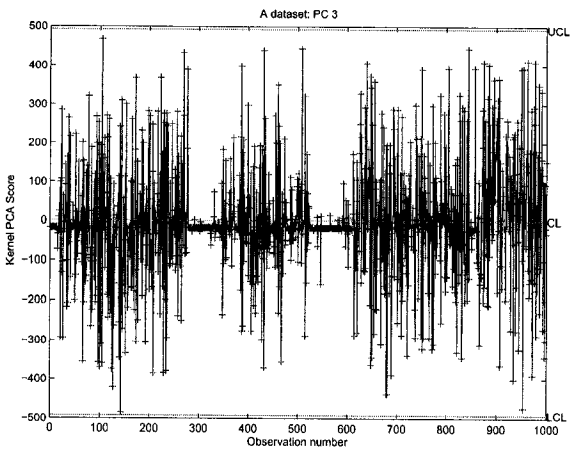

Figure 2.39: Kernel PCA Control Chart of Third Principal Component From Sample 0 to Sample 200000 of Patient 'A' File.

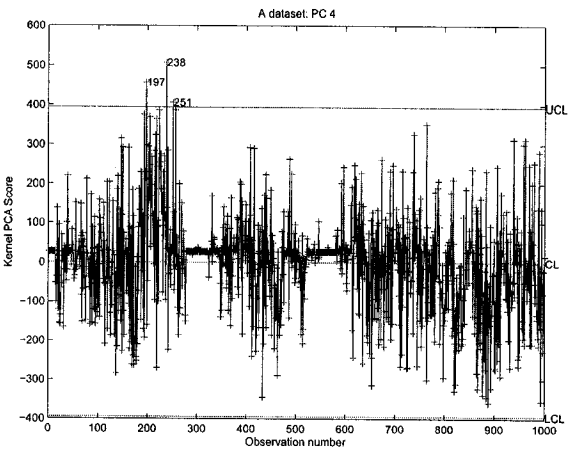

Figure 2.40: Kernel PCA Control Chart of Fourth Principal Component From Sample 0 to Sample 200000 of Patient 'A' File.

Analysis of Patient File 'B'

Fig. 2.41 through Fig. 2.51 show the charts generated by analyzing an EEG from a sample patient file. Figure 2.46 depicts a pareto diagram of the variance of each principal component relative to the sum of all principal components. Figure 2.47 depicts a scree plot of the variance due to each principal component.

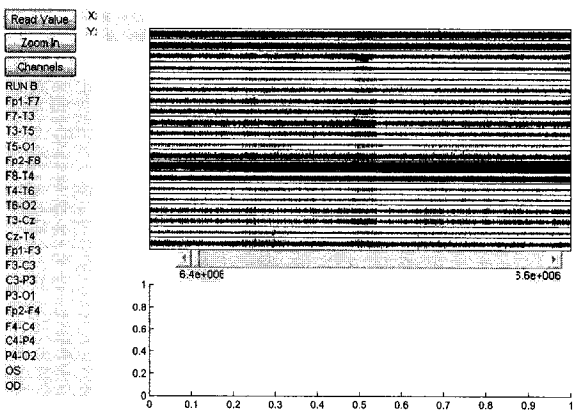

Figure 2.41: User Interface: EEG Channels Read into Software Tool.

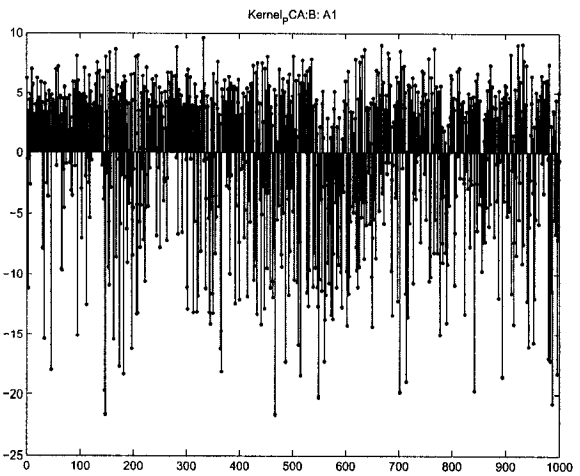

Figure 2.42: Kernel PCA: Eigenvector A1 From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

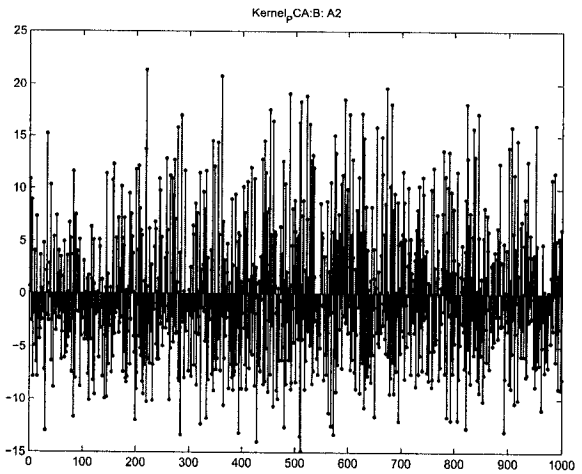

Figure 2.43: Kernel PCA: Eigenvector A2 From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

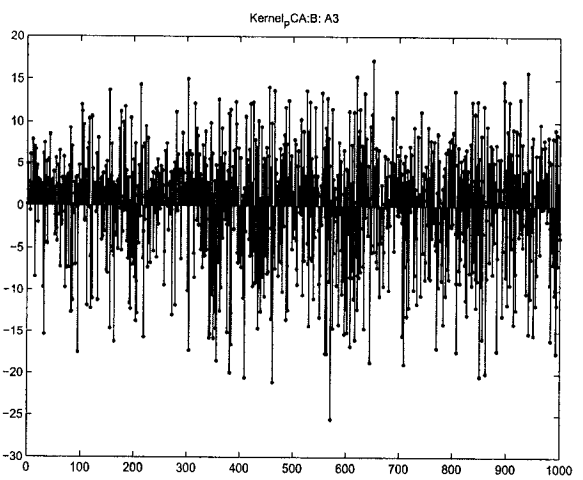

Figure 2.44: Kernel PCA: Eigenvector A3 From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

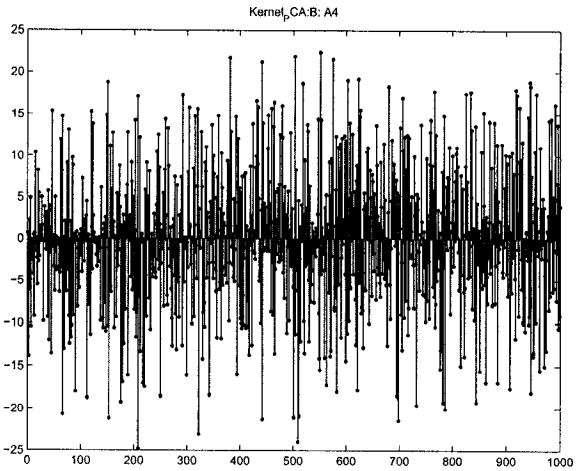

Figure 2.45: Kernel PCA: Eigenvector A4 From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

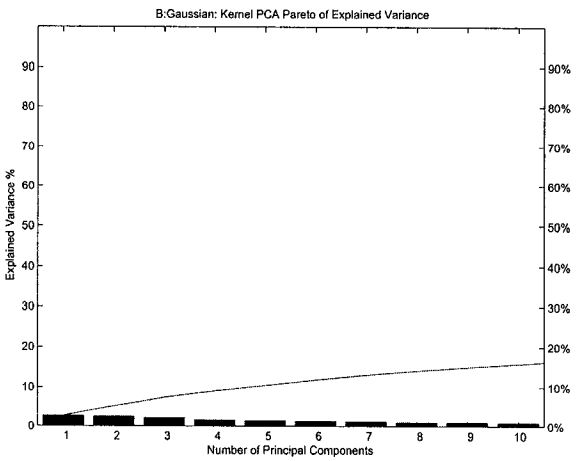

Figure 2.46: Kernel PCA: Pareto of Explained Variance From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

Analysis of Patient File 'C'

Fig. 2.52 through Fig. 2.58 show the charts generated by analyzing an EEG from a sample patient file. Figure 2.55 depicts a pareto diagram of the variance of each principal component relative to the sum of all

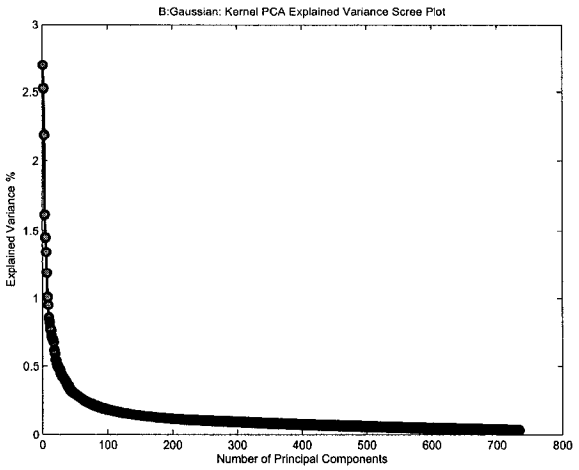

Figure 2.47: Kernel PCA: Scree Plot of Explained Variance From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

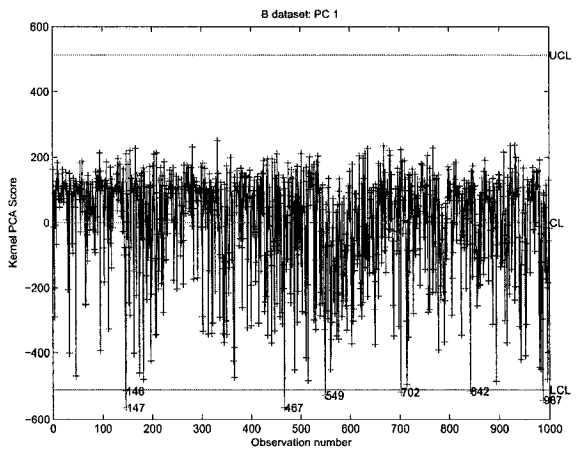

Figure 2.48: Kernel PCA Control Chart of First Principal Component From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

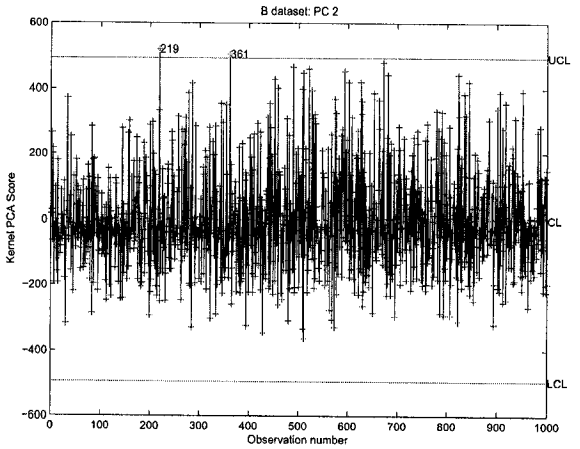

Figure 2.49: Kernel PCA Control Chart of Second Principal Component From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

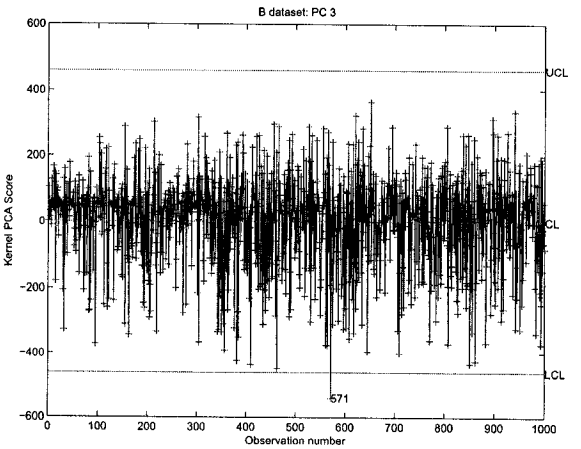

Figure 2.50: Kernel PCA Control Chart of Third Principal Component From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

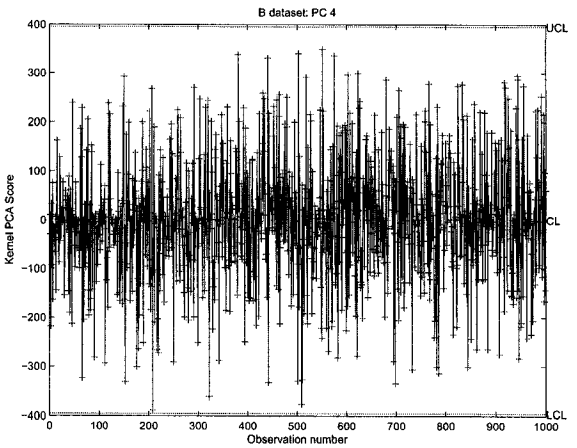

Figure 2.51: Kernel PCA Control Chart of Fourth Principal Component From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

principal components. Figure 2.56 depicts a scree plot of the variance due to each principal component.

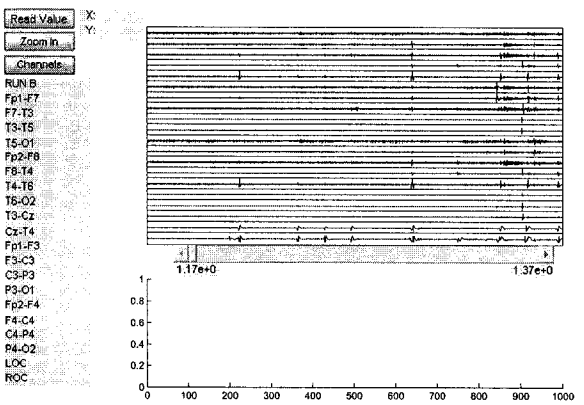

Figure 2.52: User Interface: EEG Channels Read into Software Tool.

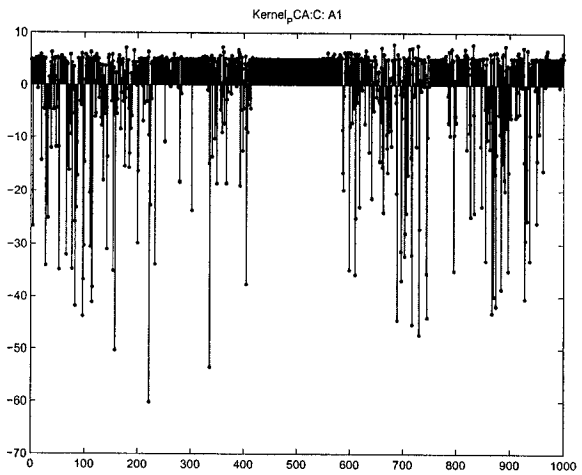

Figure 2.53: Kernel PCA: Eigenvector A1 From Sample 1.17e+006 to Sample 1.37e+006 of Patient 'C' File.

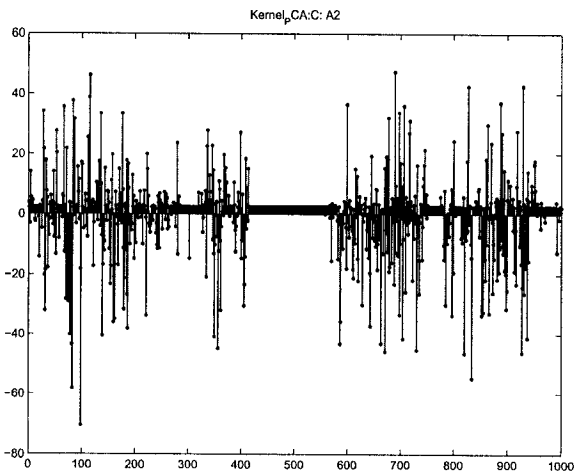

Figure 2.54: Kernel PCA: Eigenvector A2 From Sample 170000 to Sample 370000 of Patient 'C' File.

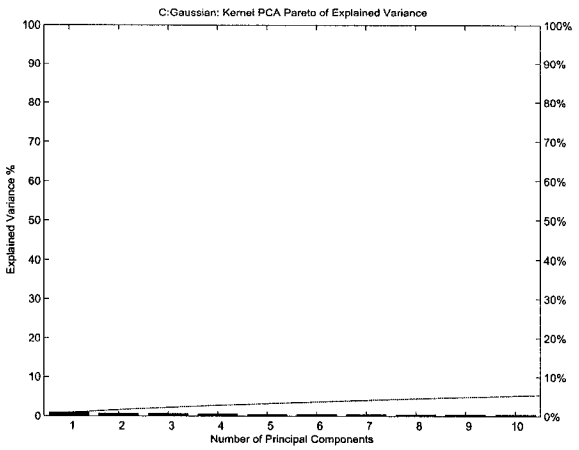

Figure 2.55: Kernel PCA: Pareto of Explained Variance From Sample 170000 to Sample 370000 of Patient 'C' File.

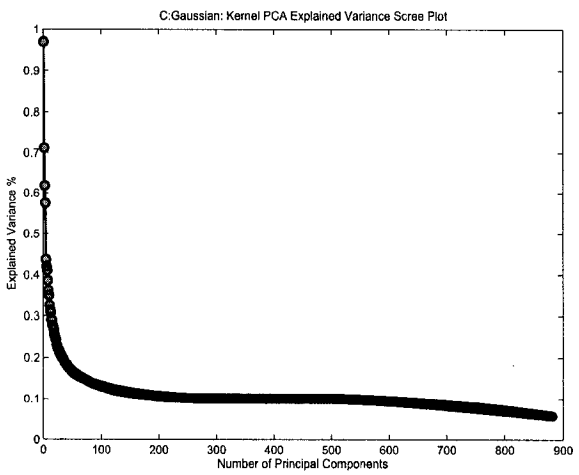

Figure 2.56: Kernel PCA: Scree Plot of Explained Variance From Sample 170000 to Sample 370000 of Patient 'C' File.

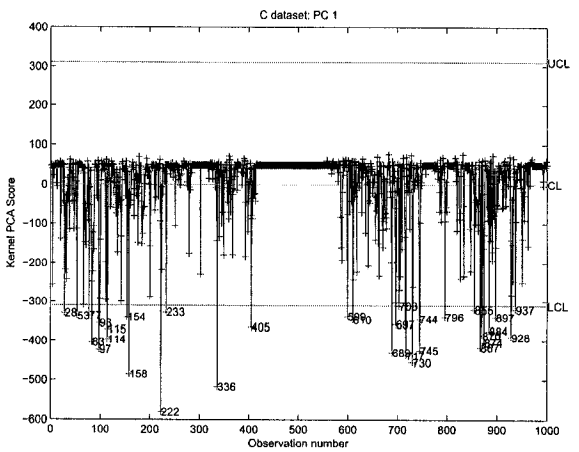

Figure 2.57: Kernel PCA Control Chart of First Principal Component From Sample 170000 to Sample 370000 of Patient 'C' File.

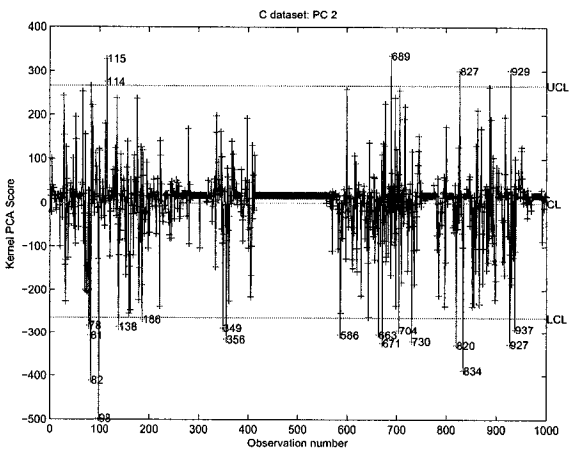

Figure 2.58: Kernel PCA Control Chart of Second Principal Component From Sample 170000 to Sample 370000 of Patient 'C' File.

2.5 Conclusions

In this section, we introduced new multivariate quality control charts by performing PCA and KPCA on EEGs. The main idea behind our proposed technique is to define new control charts used to detect outliers. The eigenvalues can be used to define control limits of our proposed control chart. The experimental results clearly show a that PCA and KPCA can be used to reduce the dimensionality of multi-dimensional data.

CHAPTER 3

Kernel Isometric Mapping of EEG Data

In this chapter, we present a robust multivariate statistical process control chart using kernel isometric feature maps (Isomap). The kernel Isomap control chart creates an Isomap kernel matrix and then performs an eigen decomposition o the mapped data. The proposed control chart is effective in the detection of outliers, and its control limits are derived from the eigen-analysis of the kernel matrix in the Hilbert feature space. Our experimental results demonstrate the effectiveness in analyzing multivariate data using the Kernel Isomap algorithm.

3.1 Introduction

In this chapter, we present a new multivariate statistical process control chart using kernel Isometric Feature Maps (Isomaps). Isomaps generalize Multi-Dimensional Scaling (MDS) to non-linear manifolds by approximating the geodesic distance along the manifold [15]. A kernel matrix is produced using Isomaps and an eigen-analysis is performed on the mapped data.

The proposed control chart is robust to outliers detection, and its control limits are derived from the eigen-analysis of the kernel Isomap matrix in the Hilbert feature space.

The remainder of the chapter is organized as follows. In Section 3.2 we propose a kernel Isomap control chart. In Section 3.3, we demonstrate the performance of the proposed multivariate control chart, using EEG data. Finally, we provide conclusions in Section 3.4.

3.2 Proposed Method

Suppose we have an input data set $X = \{\boldsymbol{x}_i : i = 1, \ldots, n\}$ where each observation $\boldsymbol{x}_i$ is a p-dimensional vector and the distribution of the data is nonlinear. Kernel Isomap algorithm consists of two main steps: the first step is to linearize the distribution of the input data by using a nonlinear mapping $\Phi : \mathbb{R}^p \rightarrow \mathcal{F}$ from the input space $\mathbb{R}^p$ to a higher-dimensional (possibly infinite-dimensional) feature space $\mathcal{F}$. The mapping Φ is defined implicitly, by specifying the form of the dot product in the feature space. In other words, given any pair of mapped data points, the dot product is defined in terms of a kernel function $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Phi(\boldsymbol{x}_i) \cdot \Phi(\boldsymbol{x}_j)$.

For the proposed method, given a data set $X = [x_1, ..., x_n]$, the geodesic distance calculation is based on a similar approach used for computing the Isomap for MDS on nonlinear manifolds. The algorithm has two main steps:

1. Construct a neighborhood graph by connecting a given point to its k-nearest neighbors, and link these neighboring points by edges with weights equal to the Euclidean distances.

2. Compute the geodesic distances $(g_{i,j})$ (shortest paths) between all pairs of n points in the constructed graph using Dijkstra's or Floyd's algorithm.

$G = (g_{i,j}^2)$ is the (square) geodesic distance matrix of size n x n.

The kernel matrix, $K_{Isomap} = -\frac{1}{2}(I - ee^T)G(I - ee^T)$, where $e = \frac{1}{\sqrt{n}}(1, 1, ..., 1)^T$.

Hence, compute the dot products in the feature space.

Once the kernel matrix is formed, we are ready to perform eigen-analysis on the mapped data.

In the second step, eigen-analysis in applied to the mapped data set $\Phi = \{\Phi_i : i = 1, \ldots, n\}$ in the feature space, where $\Phi_i = \Phi(\boldsymbol{x}_i)$. The second step of kernel Isomap is to apply eigen-analysis in the feature space by performing an eigendecomposition on the covariance matrix of the mapped data which is given by

$$C = \frac{1}{n-1} \sum_{i=1}^{n} \widetilde{\Phi}(\boldsymbol{x}_i)^T \widetilde{\Phi}(\boldsymbol{x}_i)$$

where $\widetilde{\Phi}(\boldsymbol{x}_i) = \Phi(\boldsymbol{x}_i) - (1/n) \sum_{i=1}^{n} \Phi(\boldsymbol{x}_i)$ is the centered mapped data.

The eigenvectors of C are given by

$$\boldsymbol{v} = \frac{1}{\mu} C\boldsymbol{v} = \sum_{i=1}^{n} \widetilde{\Phi}(\boldsymbol{x}_i) \left(\frac{1}{\mu(n-1)} \widetilde{\Phi}(\boldsymbol{x}_i)^T \boldsymbol{v} \right) = \sum_{i=1}^{n} \alpha_i \widetilde{\Phi}(\boldsymbol{x}_i),$$

where $\alpha_i = (\widetilde{\Phi}(\boldsymbol{x}_i)^T \boldsymbol{v})/(\mu(n-1))$. In other words, an eigenvector of C is a linear combination of $\{\widetilde{\Phi}(\boldsymbol{x}_i)\}$. Taking the dot product of $\widetilde{\Phi}(\boldsymbol{x}_j)$ with $\boldsymbol{v}$ yields

$$\widetilde{\Phi}(\boldsymbol{x}_j) \cdot \boldsymbol{v} = \sum_{i=1}^{n} \alpha_i \widetilde{\Phi}(\boldsymbol{x}_i) \cdot \widetilde{\Phi}(\boldsymbol{x}_j) = \sum_{i=1}^{n} \alpha_i \widetilde{K}_{ij},$$

which implies that $\mu(n-1)\alpha_j = \sum_{i=1}^{n} \alpha_i \widetilde{K}_{ij}$. Hence

$$\widetilde{K}\boldsymbol{\alpha} = \tilde{\mu}\boldsymbol{\alpha},$$

where $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$ and $\tilde{\mu} = \mu(n-1)$. That is, $\boldsymbol{\alpha}$ is an eigenvector of $\widetilde{K}$. If the eigenvectors of C are orthonormal (i.e. $\boldsymbol{v}^T\boldsymbol{v} = 1$) then

$$\begin{aligned} 1 = \boldsymbol{v}^T\boldsymbol{v} &= \sum_{i,j=1}^{n} \alpha_i \alpha_j \widetilde{\Phi}(\boldsymbol{x}_i) \cdot \widetilde{\Phi}(\boldsymbol{x}_j) = \sum_{i,j=1}^{n} \alpha_i \alpha_j \widetilde{K}_{ij} \\ &= \boldsymbol{\alpha}^T \widetilde{K} \boldsymbol{\alpha} = \mu(n-1)\boldsymbol{\alpha}^T\boldsymbol{\alpha} \end{aligned}$$

and hence $\|\boldsymbol{\alpha}\| = 1/\sqrt{\mu(n-1)}$.

The main algorithmic step of the proposed kernel Isomap chart as shown in Table 3.1.

Assuming we want $\pm 3\sigma$ confidence intervals, the upper control limit (UCL), the center line (CL), and the lower control limit (LCL) of the kernel Isomap chart are

$$UCL = +3\sqrt{\mu_r}$$

$$CL = 0$$

$$LCL = -3\sqrt{\mu_r}$$

3.3 Experimental Results

In this section, we demonstrate the effectiveness of Kernel Isomap by repeating the analysis on the same three EEGs and at the same intervals using Kernel Isomap. It can be seen that Kernel Isomap control charts detected outliers and the dimensionality was reduced. The results are summarized in the following subsections.

[h]

Algorithm: Kernel Isomap Control Chart

1. Construct the squared geodesic distance matrix, $G_{i,j} = \tilde{D}^2(x_i, x_j)$.
2. $e = 1/sqrt(n) * ones(n)'$.
3. Construct the kernel Isomap matrix, $K_{ij} = -\frac{1}{2}(I - ee^T)G(I - ee^T)$.
4. $K = (K_{ij})$ of the mapped data: $K_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Phi(\boldsymbol{x}_i) \cdot \Phi(\boldsymbol{x}_j)$.
5. Construct the kernel matrix $\widetilde{K} = HKH$ of the centered mapped data, where $H = I - J/n$ the centering matrix is defined in terms of the identity matrix I and the matrix of all ones J.
6. Find the largest p eigenvectors $\boldsymbol{\alpha}_r$ $(r = 1, \ldots, p)$ of $\widetilde{K}$ and their corresponding eigenvalues $\tilde{\mu}_r$.
7. Given a test point $\boldsymbol{x}$ with image $\Phi(\boldsymbol{x})$, compute the projections onto the eigenvectors $\boldsymbol{v}_r$ given by the equation

$$\boldsymbol{v}_r \cdot \widetilde{\Phi}(\boldsymbol{x}) = \frac{1}{\sqrt{(n-1)}} \sum_{i=1}^{n} \alpha_i \widetilde{\Phi}(\boldsymbol{x}_i) \cdot \widetilde{\Phi}(\boldsymbol{x})$$

Table 3.1: Algorithmic steps for Kernel Isomap.

Analysis of Patient File 'A'

Fig. 3.1 through Fig. 3.11 show the charts generated by analyzing an EEG from a sample patient file. Figure 3.6 depicts a pareto diagram of the variance of each principal component relative to the sum of all principal components. Figure 3.7 depicts a scree plot of the variance due to each principal component. Typically, the knee of the scree plot is used to determine how many principal components will be used to represent the data.

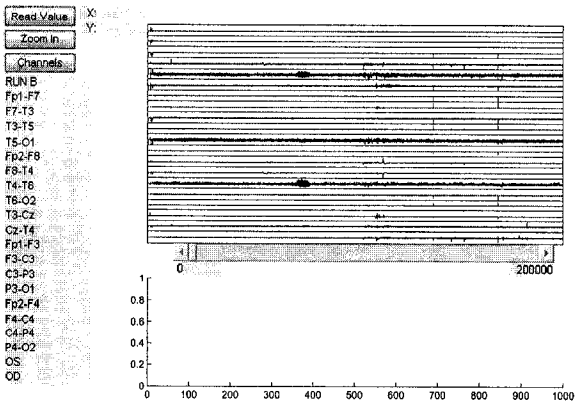

Figure 3.1: User Interface: EEG Channels Read into Software Tool.

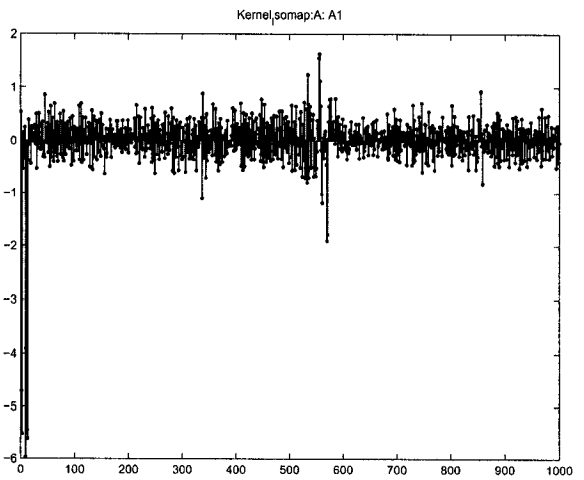

Figure 3.2: Kernel Kernel Isomap: Eigenvector A1 From Sample 0 to Sample 200000 of Patient 'A' File.

Analysis of Patient File 'B'

Fig. 3.12 through Fig. 3.22 show the charts generated by analyzing an EEG from a sample patient file. Figure 3.17 depicts a pareto diagram of the variance of each principal component relative to the sum of all principal components. Figure 3.18 depicts a scree plot of the variance due to each principal component.

As can be seen from the scree plot and pareto diagram, each principal component accounts for more variance than did kernel PCA. Of course, the results will be seen when the outliers/events are classified

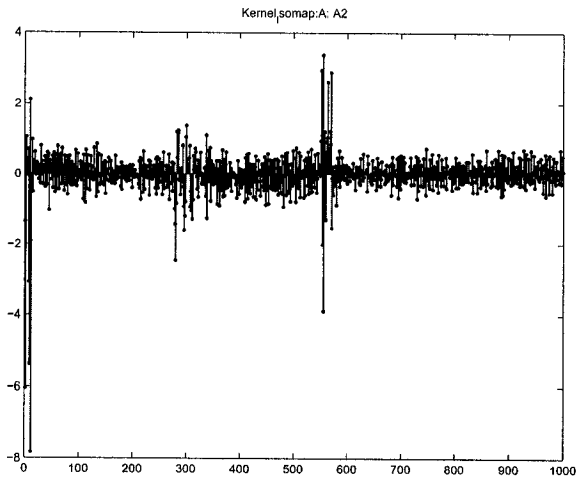

Figure 3.3: Kernel Isomap: Eigenvector A2 From Sample 0 to Sample 200000 of Patient 'A' File.

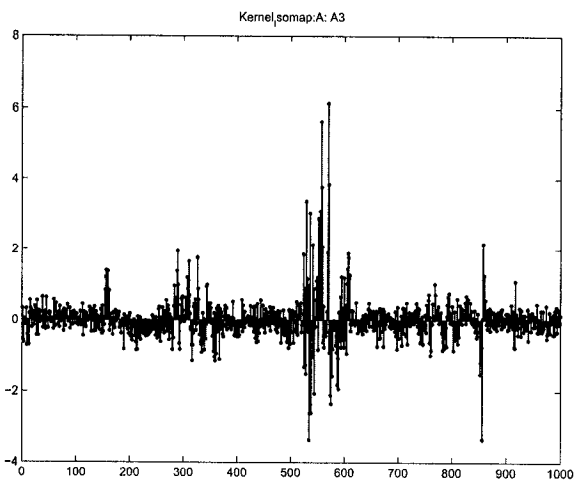

Figure 3.4: Kernel Isomap: Eigenvector A3 From Sample 0 to Sample 200000 of Patient 'A' File.

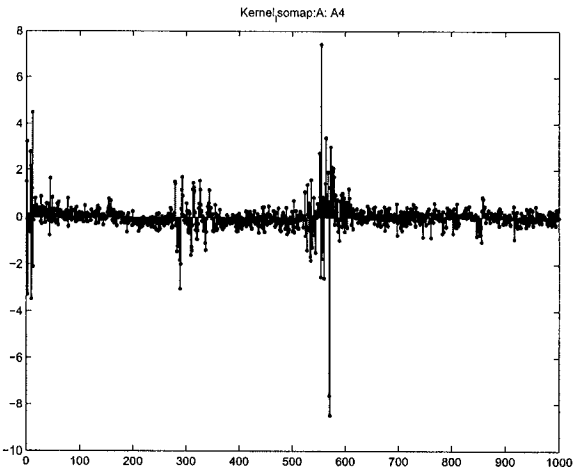

Figure 3.5: Kernel Isomap: Eigenvector A4 From Sample 0 to Sample 200000 of Patient 'A' File.

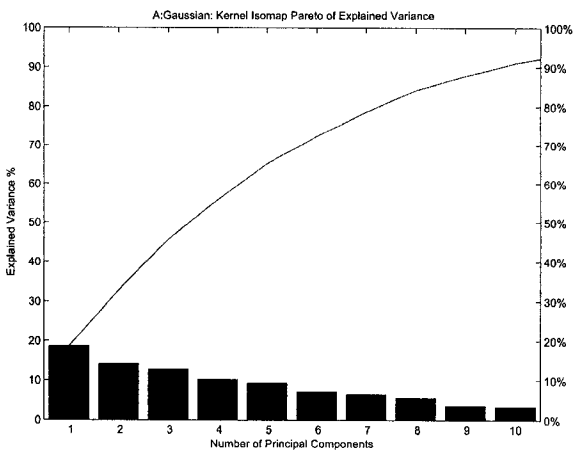

Figure 3.6: Kernel Isomap: Pareto of Explained Variance From Sample 0 to Sample 200000 of Patient 'A' File.

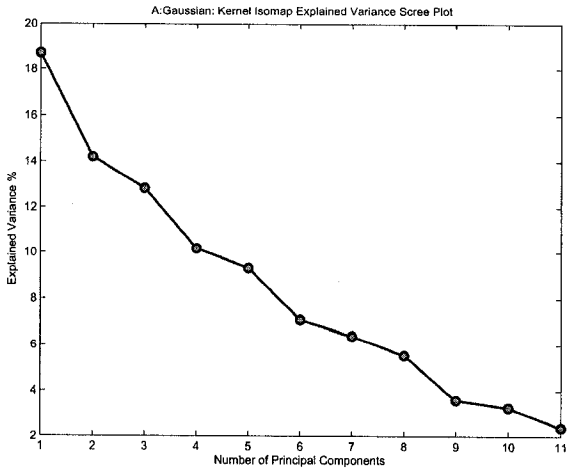

Figure 3.7: Kernel Isomap: Scree Plot of Explained Variance From Sample 0 to Sample 200000 of Patient 'A' File.

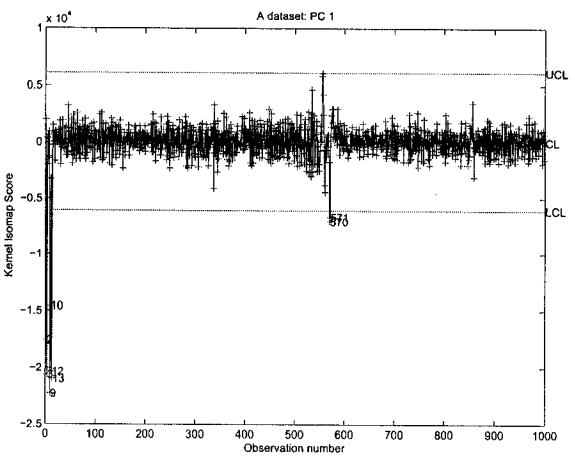

Figure 3.8: Kernel Isomap Control Chart of First Principal Component From Sample 0 to Sample 200000 of Patient 'A' File.

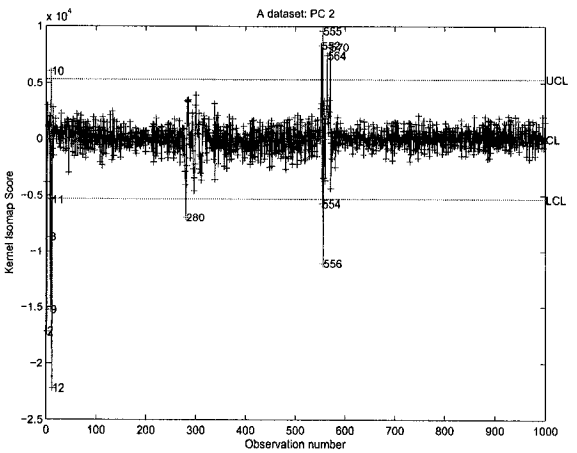

Figure 3.9: Kernel Isomap Control Chart of Second Principal Component From Sample 0 to Sample 200000 of Patient 'A' File.

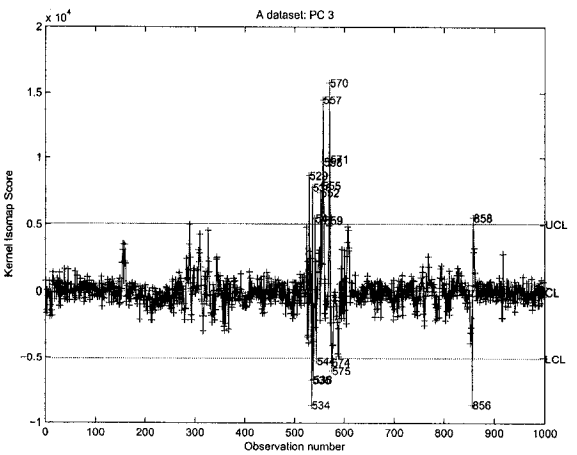

Figure 3.10: Kernel Isomap Control Chart of Third Principal Component From Sample 0 to Sample 200000 of Patient 'A' File.

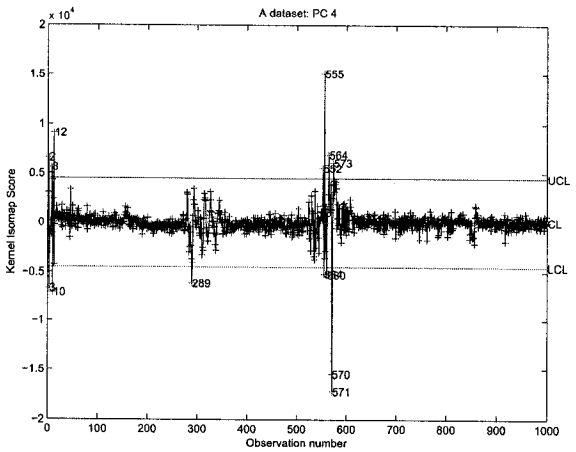

Figure 3.11: Kernel Isomap Control Chart of Fourth Principal Component From Sample 0 to Sample 200000 of Patient 'A' File.

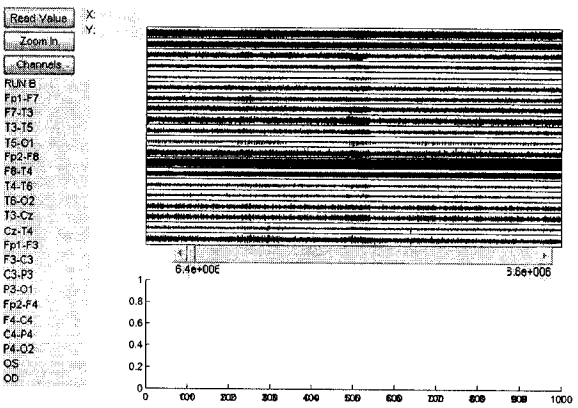

Figure 3.12: User Interface: EEG Channels Read into Software Tool.

later on. Some of the eigenvectors and control charts are presented, for comparison purposes with PCA and kernel PCA.

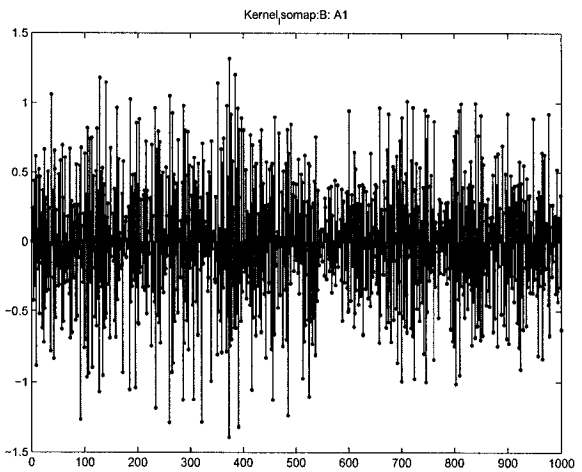

Figure 3.13: Kernel Isomap: Eigenvector A1 From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'A' File.

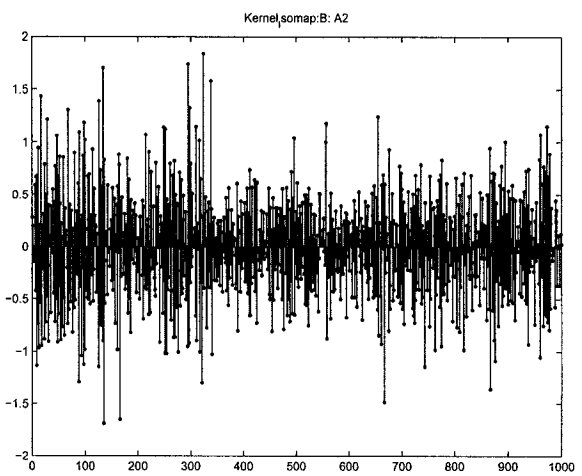

Figure 3.14: Kernel Isomap: Eigenvector A2 From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'A' File.

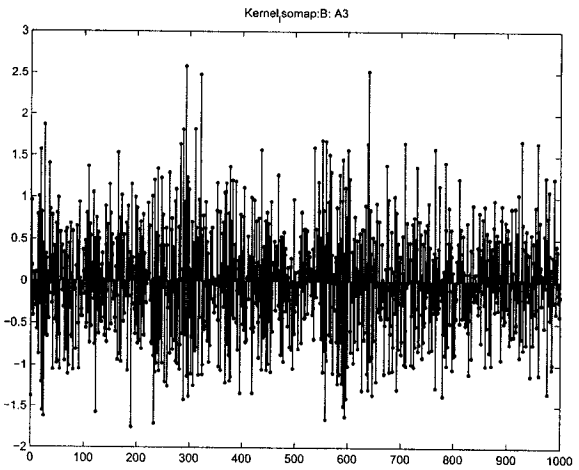

Figure 3.15: Kernel Isomap: Eigenvector A3 From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'A' File.

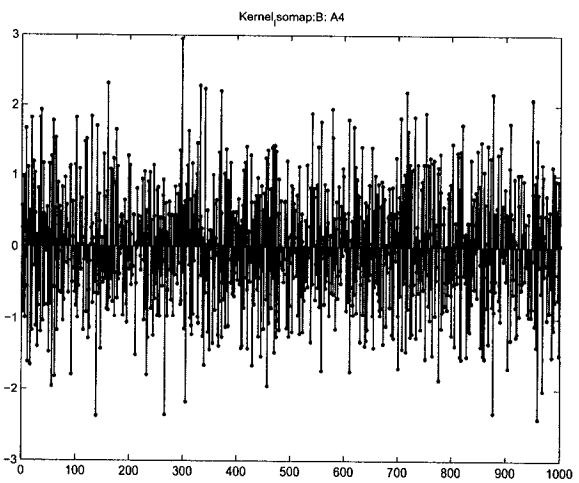

Figure 3.16: Kernel Isomap: Eigenvector A4 From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'A' File.

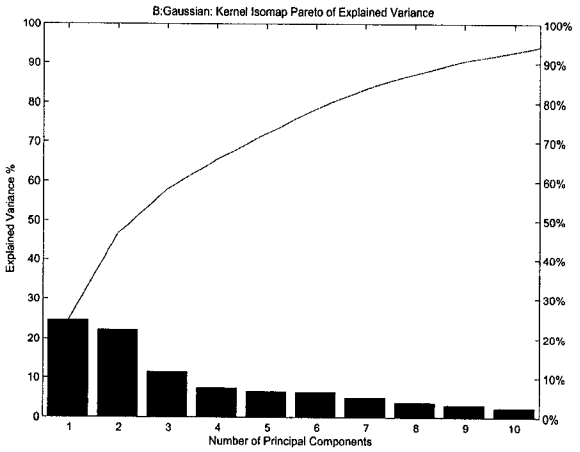

Figure 3.17: Kernel Isomap: Pareto of Explained Variance From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

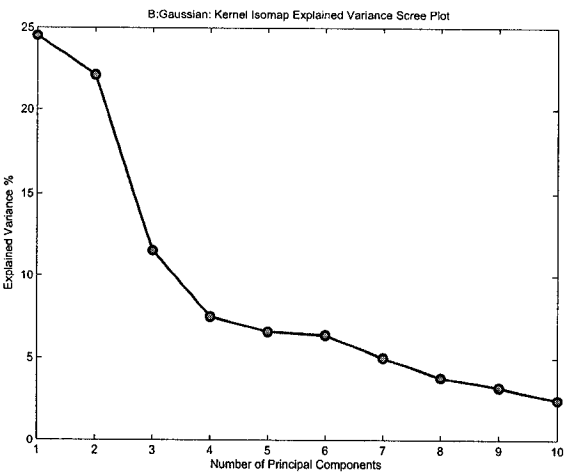

Figure 3.18: Kernel Isomap: Scree Plot of Explained Variance From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

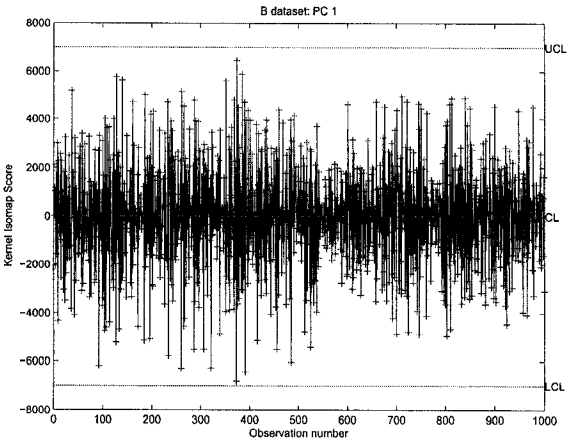

Figure 3.19: Kernel Isomap Control Chart of First Principal Component From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

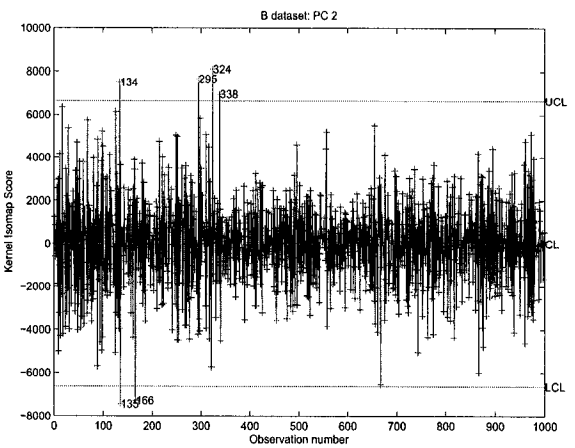

Figure 3.20: Kernel Isomap Control Chart of Second Principal Component From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

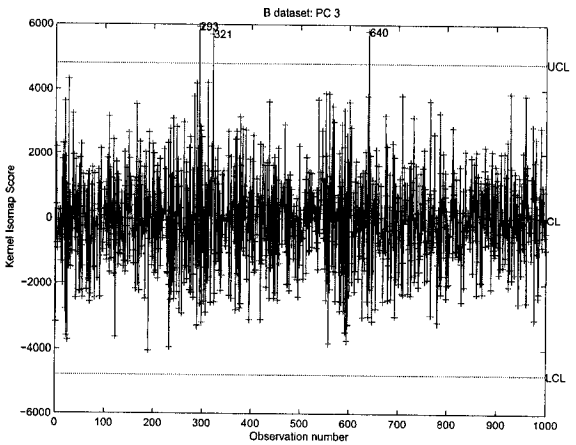

Figure 3.21: Kernel Isomap Control Chart of Third Principal Component From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

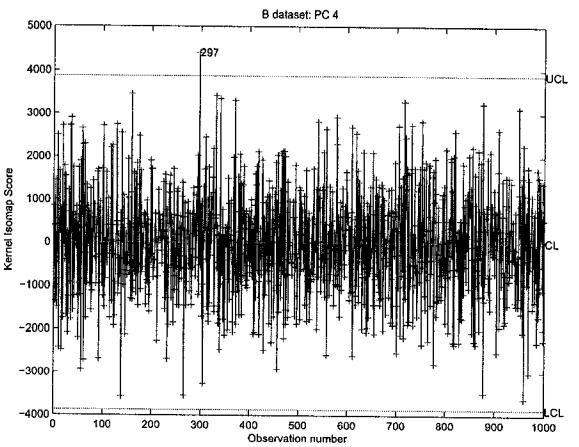

Figure 3.22: Kernel Isomap Control Chart of Fourth Principal Component From Sample 6.4e+006 to Sample 6.6e+006 of Patient 'B' File.

Analysis of Patient File 'C'

Fig. 3.23 through Fig. 3.29 show the charts generated by analyzing an EEG from a sample patient file. Figure 3.26 depicts a pareto diagram of the variance of each principal component relative to the sum of all principal components. Figure 3.27 depicts a scree plot of the variance due to each principal component.

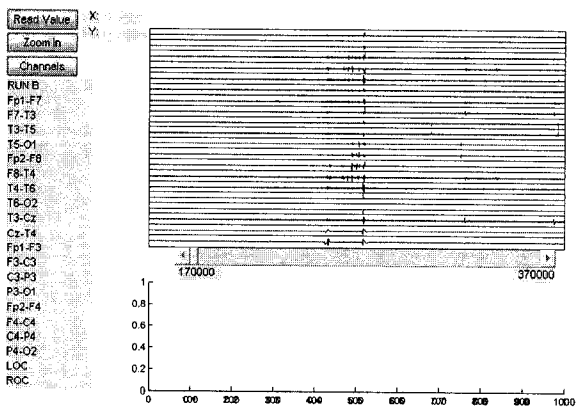

Figure 3.23: User Interface: EEG Channels Read into Software Tool.

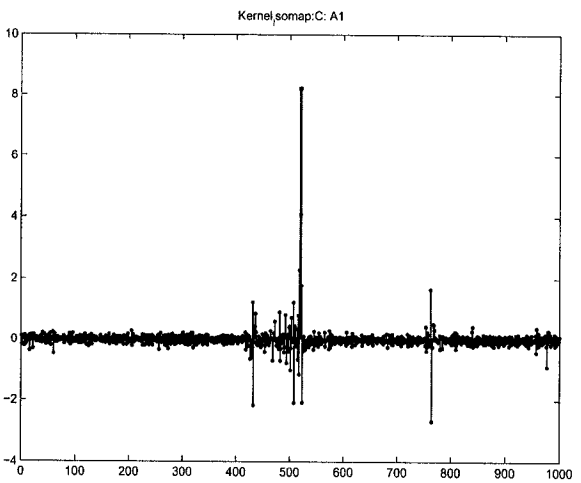

Figure 3.24: Kernel Isomap: Eigenvector A1 From Sample 1.17e+006 to Sample 1.37e+006 of Patient 'C' File.

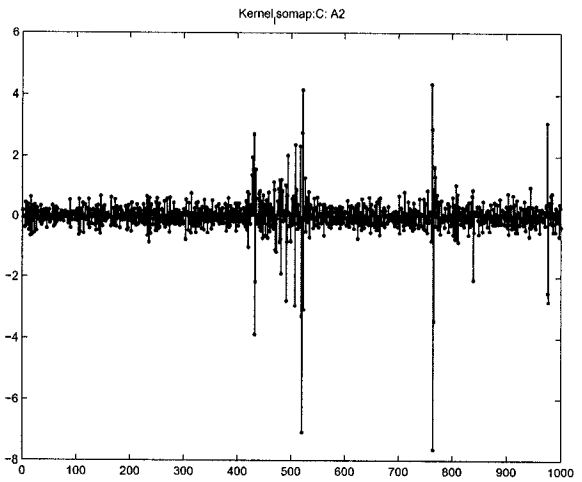

Figure 3.25: Kernel Isomap: Eigenvector A2 From Sample 170000 to Sample 370000 of Patient 'C' File.

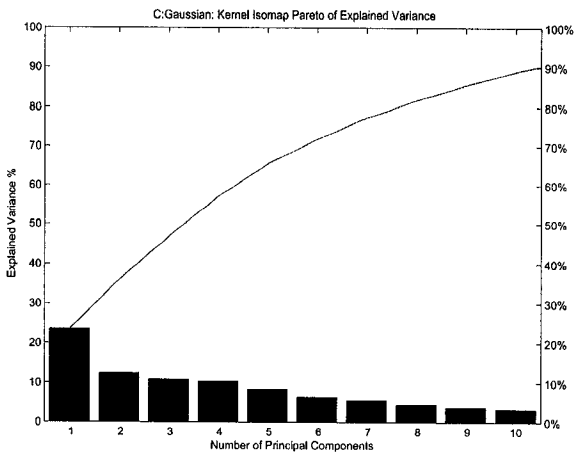

Figure 3.26: Kernel Isomap: Pareto of Explained Variance From Sample 170000 to Sample 370000 of Patient 'C' File.

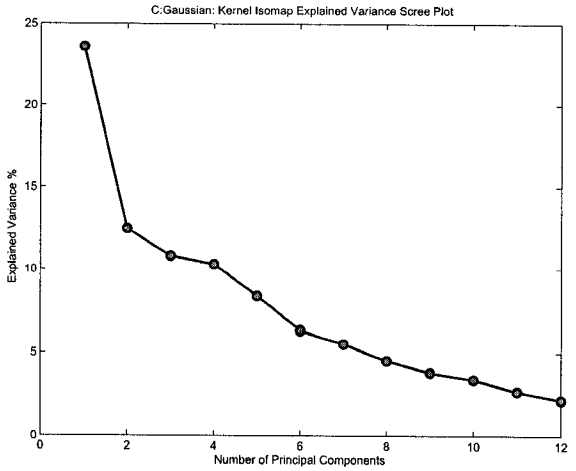

Figure 3.27: Kernel Isomap: Scree Plot of Explained Variance From Sample 170000 to Sample 370000 of Patient 'C' File.

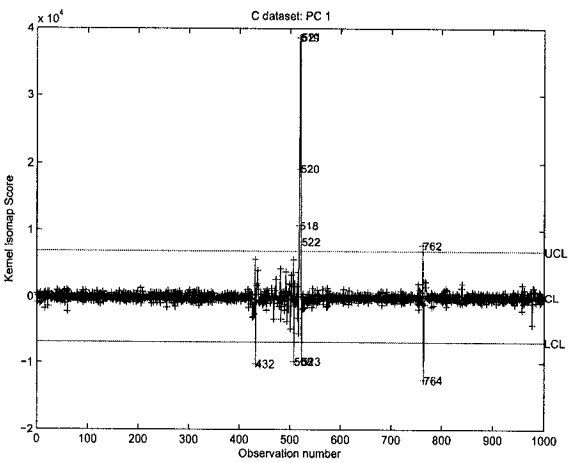

Figure 3.28: Kernel Isomap Control Chart of First Principal Component From Sample 170000 to Sample 370000 of Patient 'C' File.

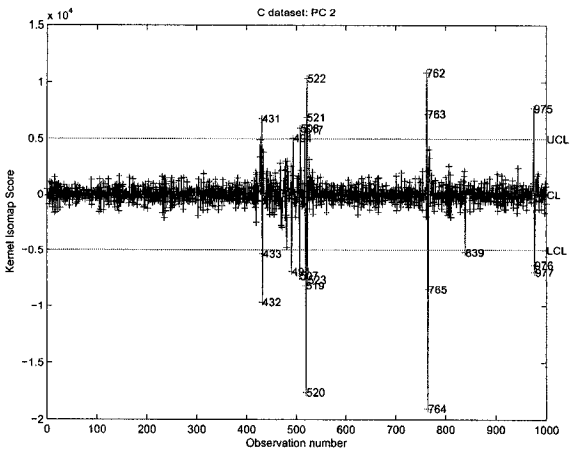

Figure 3.29: Kernel Isomap Control Chart of Second Principal Component From Sample 170000 to Sample 370000 of Patient 'C' File.

3.4 Conclusions

In this chapter, we introduced a new multivariate control chart by using the concept of kernel Isomap eigen-analysis. The core idea behind our proposed technique is to create an Isomap kernel matrix prior to projecting the data into a higher dimension Hilbert space and then extracting the eigenvalues and eigenvectors of the kernel matrix. The experimental results clearly show dimensionality reduction is successfully achieved by the use of kernel Isomaps.

CHAPTER 4

Kernel Locally Linear Embedding Algorithm for Outliers Detection

In this chapter, we introduce a new multivariate statistical process control chart for outlier detection using a kernel local linear embedding algorithm [16]. The proposed control chart is effective in the detection of outliers, and its control limits are derived from the eigen-analysis of the kernel matrix in the Hilbert feature space. Our experimental results show the much improved performance of the proposed control chart in comparison with existing multivariate monitoring and controlling charts.

4.1 Introduction

Traditional process monitoring consists of measuring and controlling several process variables at the same time [2]. It is increasing difficult to determine the root cause of defects if multiple process variables exhibit outliers or process deviations at the same moment in time. Multivariate quality control methods overcome this disadvantage by monitoring the interactions of several process variables simultaneously and determining hidden factors using dimensionality reduction [5]. The use of multivariate statistical process control is also facilitated by the proliferation of sensor data that is typically complex, high-dimensional and generally correlated. Complex processes can be monitored the stability evaluated, using multivariate statistical process control techniques.

There are typically two phases in establishing multivariate control charts. The data collected in phase I are used to establish the control limits for phase II.

In recent years, a variety of statistical quality control methods have been proposed to monitor multivariate data including Hotelling's T^2-statistic chart [2], and the principal component analysis control chart based on principal component analysis [14]. These control charts are widely used in industry, particularly in assembly operations and chemical process control [5]. The T^2 statistic is, however, vulnerable to outliers and in order to obtain significantly good results, both the mean and the covariance matrix must be robustly estimated [17–20]. Also, principal component analysis is very sensitive to outliers [5].

In this chapter, we present a new multivariate statistical process control chart using kernel locally linear embedding. Locally linear embedding (LLE) is a recently proposed unsupervised procedure for mapping high-dimensional data nonlinearly to a lower-dimensional space [22]. The basic idea of LLE is that of global minimization of the reconstruction error of the set of all local neighborhoods in the data set. The proposed kernel LLE control chart is robust to outliers detection, and its control limits are derived from the eigen-analysis of the kernel LLE matrix in the Hilbert feature space.

The remainder of the chapter is organized as follows. In Section 4.2, we propose a kernel LLE control chart. In Section 4.3, we demonstrate through experimental results that the performance of the proposed multivariate control chart has greatly been improved in comparison with existing monitoring and controlling charts. Finally, we conclude in Section 4.4.

4.2 Proposed Method

LLE algorithm aims at finding an embedding that preserves the local geometry in the neighborhood of each data point. First, we build a sparse matrix of local predictive weights $W_{i,j}$, such that $\sum_j W_{i,j} = 1$, $W_{i,j} = 0$ if x_j is not a k-nearest neighbor of x_i and then $\sum_j (W_{i,j}x_j - x_i)^2$ is minimized to create the matrix $M = (I - W)^T(I - W)$. Then we define the kernel matrix $K = \lambda_{max}I - M$, where λ_{max} is the maximum eigenvalue of M.

Suppose we have an input data set $X = \{\boldsymbol{x}_i : i = 1, \ldots, n\}$ where each observation $\boldsymbol{x}_i$ is a p-dimensional vector. Kernel PCA algorithm [23, 24] consists of two main steps: the first step is to linearize the distribution of the input data by using a non-linear mapping $\Phi : \mathbb{R}^p \rightarrow \mathcal{F}$ from the input space $\mathbb{R}^p$ to a higher-dimensional (possibly infinite-dimensional) feature space $\mathcal{F}$. The mapping Φ is defined implicitly, by specifying the form of the dot product in the feature space. In other words, given any pair of mapped

data points, the dot product is defined in terms of a kernel function $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Phi(\boldsymbol{x}_i) \cdot \Phi(\boldsymbol{x}_j)$.

In the second step, eigen-analysis in applied to the mapped data set $\Phi = \{\Phi_i : i = 1, \ldots, n\}$ in the feature space, where $\Phi_i = \Phi(\boldsymbol{x}_i)$. The second step of kernel PCA is to perform an eigen-decomposition of the covariance matrix of the mapped data which is given by

$$C = \frac{1}{n-1} \sum_{i=1}^{n} \widetilde{\Phi}(\boldsymbol{x}_i)^T \widetilde{\Phi}(\boldsymbol{x}_i)$$

where $\widetilde{\Phi}(\boldsymbol{x}_i) = \Phi(\boldsymbol{x}_i) - (1/n) \sum_{i=1}^{n} \Phi(\boldsymbol{x}_i)$ is the centered mapped data.

The eigenvectors of C are given by

$$\boldsymbol{v} = \frac{1}{\mu} C\boldsymbol{v} = \sum_{i=1}^{n} \widetilde{\Phi}(\boldsymbol{x}_i) \left(\frac{1}{\mu(n-1)} \widetilde{\Phi}(\boldsymbol{x}_i)^T \boldsymbol{v} \right) = \sum_{i=1}^{n} \alpha_i \widetilde{\Phi}(\boldsymbol{x}_i),$$

where $\alpha_i = (\widetilde{\Phi}(\boldsymbol{x}_i)^T \boldsymbol{v})/(\mu(n-1))$. In other words, an eigenvector of C is a linear combination of $\{\widetilde{\Phi}(\boldsymbol{x}_i)\}$. Taking the dot product of $\widetilde{\Phi}(\boldsymbol{x}_j)$ with $\boldsymbol{v}$ yields

$$\widetilde{\Phi}(\boldsymbol{x}_j) \cdot \boldsymbol{v} = \sum_{i=1}^{n} \alpha_i \widetilde{\Phi}(\boldsymbol{x}_i) \cdot \widetilde{\Phi}(\boldsymbol{x}_j) = \sum_{i=1}^{n} \alpha_i \widetilde{K}_{ij},$$

which implies that $\mu(n-1)\alpha_j = \sum_{i=1}^{n} \alpha_i \widetilde{K}_{ij}$. Hence

$$\widetilde{K}\boldsymbol{\alpha} = \tilde{\mu}\boldsymbol{\alpha},$$

where $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)$ and $\tilde{\mu} = \mu(n-1)$. That is, $\boldsymbol{\alpha}$ is an eigenvector of $\widetilde{K}$. If the eigenvectors of C are orthonormal (i.e. $\boldsymbol{v}^T\boldsymbol{v} = 1$) then

$$\begin{aligned} 1 = \boldsymbol{v}^T\boldsymbol{v} &= \sum_{i,j=1}^{n} \alpha_i \alpha_j \widetilde{\Phi}(\boldsymbol{x}_i) \cdot \widetilde{\Phi}(\boldsymbol{x}_j) = \sum_{i,j=1}^{n} \alpha_i \alpha_j \widetilde{K}_{ij} \\ &= \boldsymbol{\alpha}^T \widetilde{K} \boldsymbol{\alpha} = \mu(n-1)\boldsymbol{\alpha}^T\boldsymbol{\alpha} \end{aligned}$$

and hence $\|\boldsymbol{\alpha}\| = 1/\sqrt{\mu(n-1)}$.

The main algorithmic step of our proposed kernel LLE chart as shown in Table 4.1. The kernel LLE algorithm is based on the concepts of LLE and kernel PCA.

Assuming we want $\pm 3\sigma$ confidence intervals, the upper control limit (UCL), the center line (CL), and the lower control limit (LCL) of the kernel LLE chart are

$$UCL = +3\sqrt{\tilde{\mu}_r}$$

Algorithm: Kernel LLE Control Chart

1. Construct a sparse matrix of local predictive weights $W_{i,j}$, such that $\sum_j W_{i,j} = 1$, $W_{i,j} = 0$ if x_j is not a k-nearest neighbor of x_i and $\sum_j (W_{i,j}x_j - x_i)^2$ is minimized.
2. Construct the LLE matrix, $M = (I - W)^T(I - W)$.
3. Construct the kernel matrix $K = \lambda_{max}I - M$.
4. $K = (K_{ij})$ of the mapped data: $K_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Phi(\boldsymbol{x}_i) \cdot \Phi(\boldsymbol{x}_j)$.
5. Construct the kernel matrix $\widetilde{K} = HKH$ of the centered mapped data, where $H = I - J/n$ the centering matrix is defined in terms of the identity matrix I and the matrix of all ones J.
6. Find the largest p eigenvectors $\boldsymbol{\alpha}_r$ $(r = 1, \ldots, p)$ of $\widetilde{K}$ and their corresponding eigenvalues $\tilde{\mu}_r$.
7. Given a test point $\boldsymbol{x}$ with image $\Phi(\boldsymbol{x})$, compute the projections onto the eigenvectors $\boldsymbol{v}_r$ given by the equation

$$\boldsymbol{v}_r \cdot \widetilde{\Phi}(\boldsymbol{x}) = \frac{1}{\sqrt{(n-1)}} \sum_{i=1}^{n} \alpha_i \widetilde{\Phi}(\boldsymbol{x}_i) \cdot \widetilde{\Phi}(\boldsymbol{x})$$

Table 4.1: Algorithmic steps for Kernel LLE.

$$CL = 0$$

$$LCL = -3\sqrt{\tilde{\mu}_r}$$

4.3 Experimental Results

We conducted experiments on three different data sets with known outliers. In all the experiments, the number of nearest neighbors was set to one less than the rank of the input matrix and the dimension of the output matrix was set to the number of input vectors.

4.3.1 Experiment #1: Woodmod Dataset

We tested the performance of our proposed technique on a data set $X = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{20}]^T$ (called woodmod data [20]) which contains 20 observations as shown in Table 4.2.

x_{i1}	x_{i2}	x_{i3}	x_{i4}	x_{i5}
0.5730	0.1059	0.4650	0.5380	0.8410
0.6510	0.1356	0.5270	0.5450	0.8870
0.6060	0.1273	0.4940	0.5210	0.9200
0.4370	0.1591	0.4460	0.4230	0.9920
0.5470	0.1135	0.5310	0.5190	0.9150
0.4440	0.1628	0.4290	0.4110	0.9840
0.4890	0.1231	0.5620	0.4550	0.8240
0.4130	0.1673	0.4180	0.4300	0.9780
0.5360	0.1182	0.5920	0.4640	0.8540
0.6850	0.1564	0.6310	0.5640	0.9140
0.6640	0.1588	0.5060	0.4810	0.8670
0.7030	0.1335	0.5190	0.4840	0.8120
0.6530	0.1395	0.6250	0.5190	0.8920
0.5860	0.1114	0.5050	0.5650	0.8890
0.5340	0.1143	0.5210	0.5700	0.8890
0.5230	0.1320	0.5050	0.6120	0.9190
0.5800	0.1249	0.5460	0.6080	0.9540
0.4480	0.1028	0.5220	0.5340	0.9180
0.4170	0.1687	0.4050	0.4150	0.9810
0.5280	0.1057	0.4240	0.5660	0.9090

Table 4.2: Woodmod dataset.

Each observation $\boldsymbol{x}_i = (x_{i1}, x_{i2}, x_{i3}, x_{i4}, x_{i5})$ has 5 variables which correspond respectively to:

- number of fibers per square millimeter in Springwood
- number of fibers per square millimeter in Summerwood
- fraction of Springwood
- fraction of light absorption by Springwood
- fraction of light absorption by Summerwood

The woodmod data variables are highly correlated as shown in Fig. 4.1, and hence multidimensional quality control charts should be applied.

Fig. 4.2 shows that the T^2 control chart is unable to detect outliers. Also, the principal component control chart is unable to detect outliers as depicted in Fig. 4.3. We can clearly see in Fig. 4.3 that the observations 4, 6, 8 and 19 have higher variations than the rest of the observations although they still lie within the upper and lower control limits.

The kernel LLE chart is able to detect the observations 2, 4, 6, 8, 10, 11, 12, 13 and 19 as outliers as shown in Fig. 4.4.

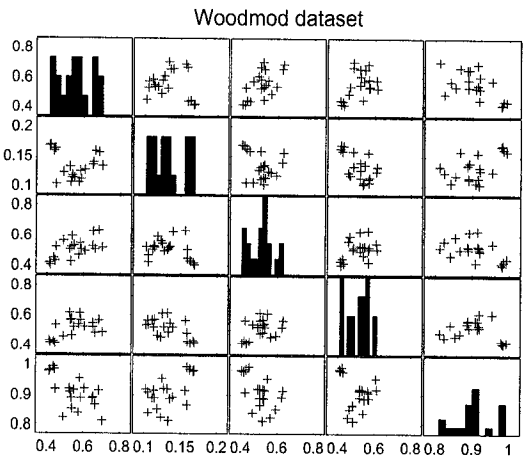

Figure 4.1: Scatter plot of the woodmod dataset.

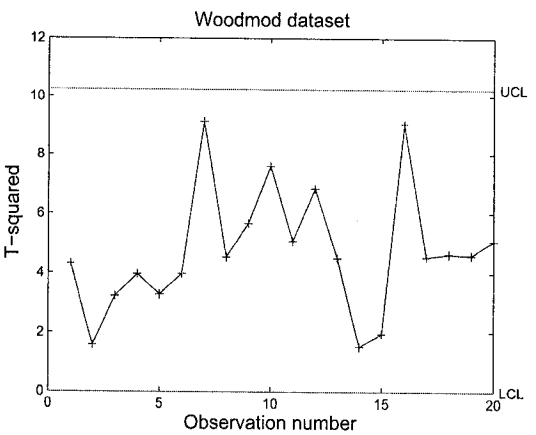

Figure 4.2: T^2 control chart.

4.3.2 Experiment #2: Stackloss Dataset

Our second analysis was performed on a dataset called Stackloss shown in Table 4.3. This dataset describes the plant oxidation of ammonia to nitric acid, and contains 21 observations, where each observation has 4 variables: rate, temperature, acid concentration, and stackloss.

The scatter plot shown in Fig. 4.5 confirms the existence of a high correlation between the variables.

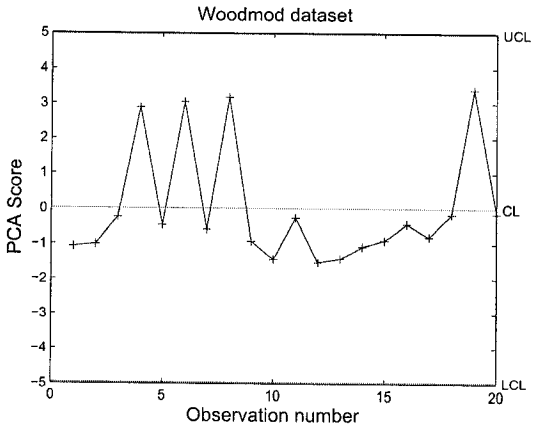

Figure 4.3: Principal component chart.

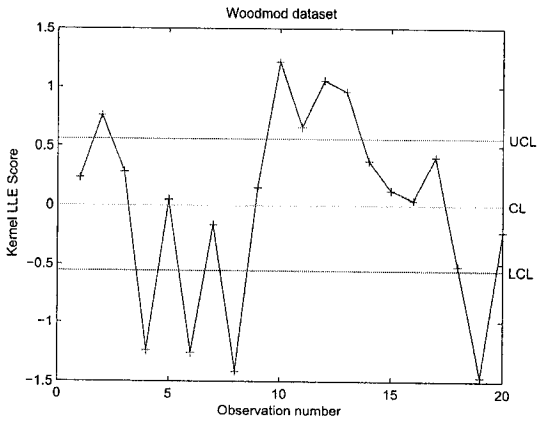

Figure 4.4: Kernel LLE chart.

The T^2 control chart displayed in Fig. 4.6 was able to identify the last observation ($m = 21$) as an outlier. The principal component chart, however, did not detect any outliers as shown in Fig. 4.7.

On the other hand, the kernel LLE chart (see Fig. 4.8) was able to identify observations 1, 2, 3, 15, 16, 17, 18, and 19 as outliers.

x_{i1}	x_{i2}	x_{i3}	x_{i4}
80.0	27.0	89.0	42.0
80.0	27.0	88.0	37.0
75.0	25.0	90.0	37.0
62.0	24.0	87.0	28.0
62.0	22.0	87.0	18.0
62.0	23.0	87.0	18.0
62.0	24.0	93.0	19.0
62.0	24.0	93.0	20.0
58.0	23.0	87.0	15.0
58.0	18.0	80.0	14.0
58.0	18.0	89.0	14.0
58.0	17.0	88.0	13.0
58.0	18.0	82.0	11.0
58.0	19.0	93.0	12.0
50.0	18.0	89.0	8.0
50.0	18.0	86.0	7.0
50.0	19.0	72.0	8.0
50.0	19.0	79.0	8.0
50.0	20.0	80.0	9.0
56.0	20.0	82.0	15.0
70.0	20.0	91.0	15.0

Table 4.3: Stackloss data set.

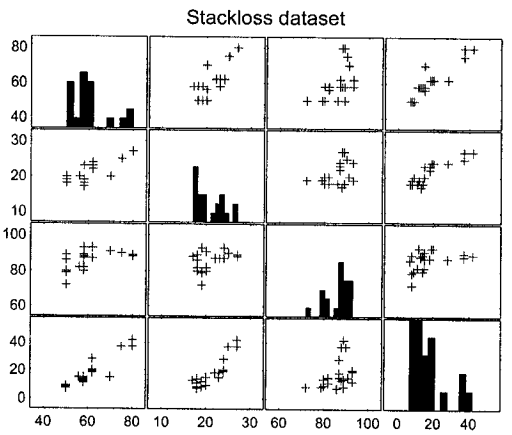

Figure 4.5: Scatter plot of Stackloss dataset.

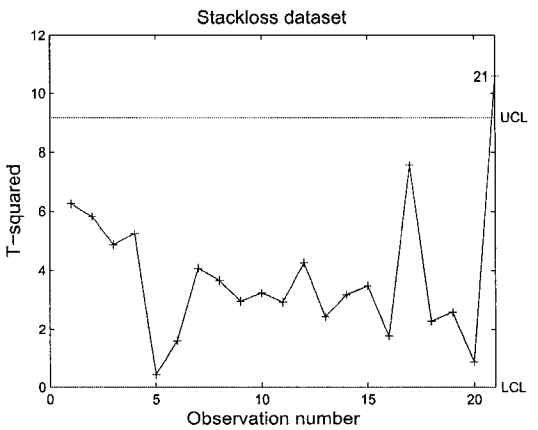

Figure 4.6: T^2 control chart

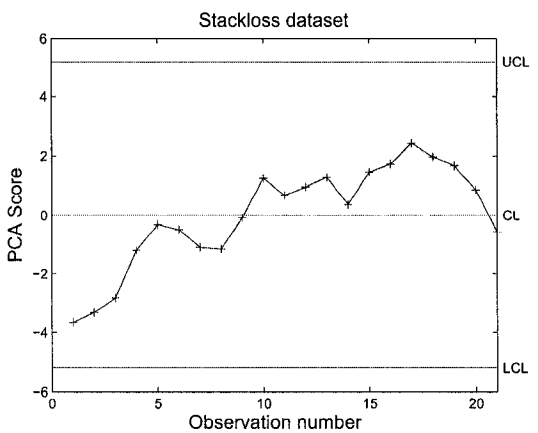

Figure 4.7: Principal component chart.

4.3.3 Experiment #3: Phosphorus Content Data

Our third analysis was performed on a dataset (Table 4.4) describing the organic and inorganic phosphorus content of the soil in comparison with the corn grown. Eighteen observations were selected where each observation has three variables: inorganic phosphorus, organic phosphorus, and plant phosphorus.

The scatter plot of the data set is displayed in Fig. 4.9. As shown in Fig. 4.10, the T^2 control chart was able to identify the observation 17 as an outlier, whereas principal component chart did not identify any

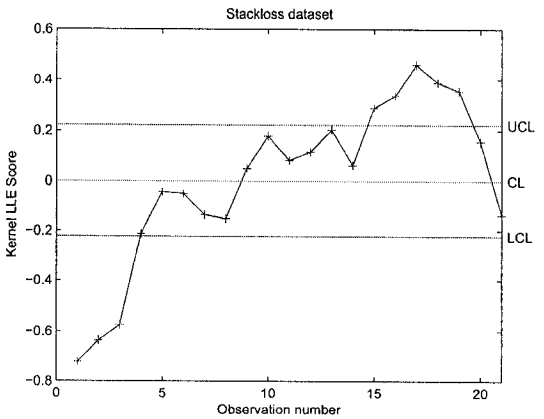

Figure 4.8: Kernel LLE chart.

x_{i1}	x_{i2}	x_{i3}
0.40	53.00	64.00
0.40	23.00	60.00
3.10	19.00	71.00
0.60	34.00	61.00
4.70	24.00	54.00
1.70	65.00	77.00
9.40	44.00	81.00
10.10	31.00	93.00
11.60	29.00	93.00
12.60	58.00	51.00
10.90	37.00	76.00
23.10	46.00	96.00
23.10	50.00	77.00
21.60	44.00	93.00
23.10	56.00	95.00
1.90	36.00	54.00
26.80	58.00	168.00
29.90	51.00	99.00

Table 4.4: Phosphorus Content data set.

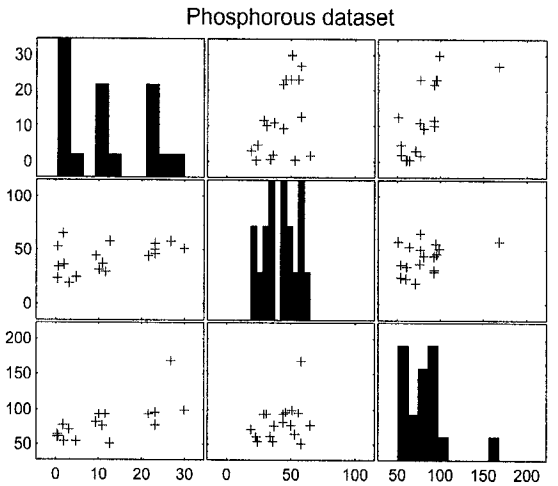

Figure 4.9: Scatter Plot of Phosphorous dataset

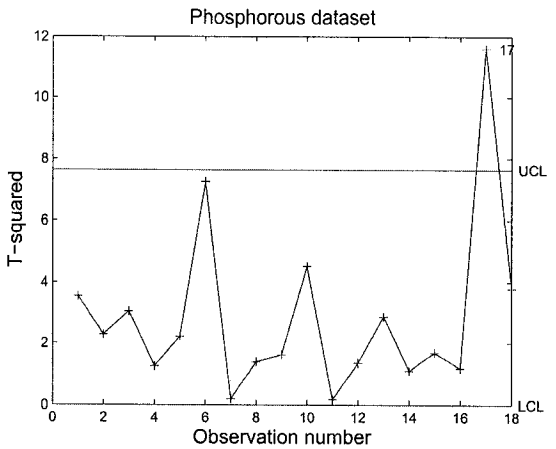

Figure 4.10: T^2 control chart

outliers as illustrated in Fig. 4.11.

Kernel LLE chart was, however, able to detect the observations 2, 3, 4, 5, 12, 15, 16, 17, and 18 as outliers as shown in Fig. 4.12.

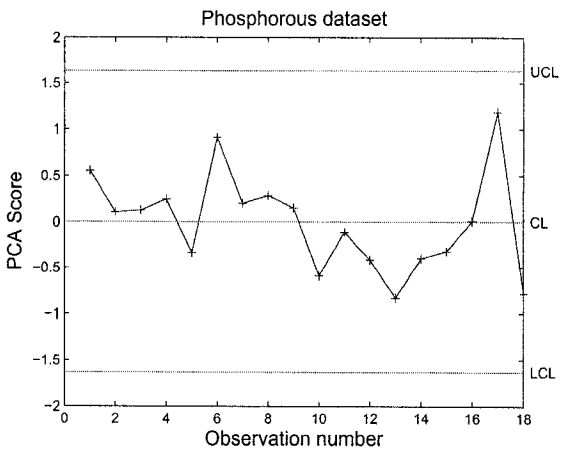

Figure 4.11: Principal component chart.

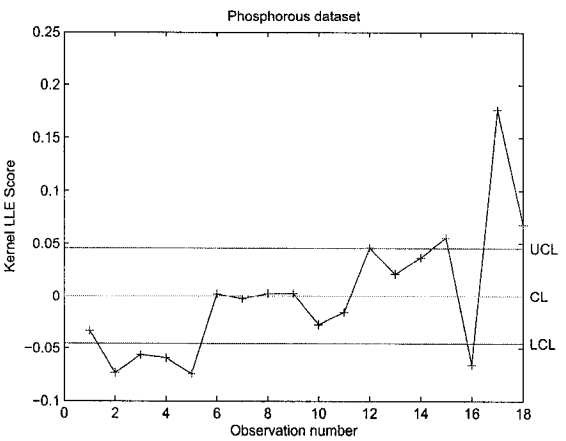

Figure 4.12: Kernel LLE chart.

4.4 Conclusions

In this chapter, we proposed a robust multivariate control chart for outliers detection using kernel locally linear embedding algorithm. The core idea behind our proposed technique is to project the data into a Hilbert space in order to extract the eigenvalues and eigenvectors of a kernel matrix. The experimental results clearly show a much improved performance of the proposed approach in comparison with the current multivariate control charts.

CHAPTER 5

Conclusions

This thesis has presented dimensionality reduction algorithms to analyze multivariate data. Our algorithms are built on the foundations of kernel matrices and dimension reduction techniques. We have demonstrated the effectiveness of the proposed methods with real EEG data.

In the next Section, the contributions made in each of the previous chapters and the concluding results drawn from the associated research are presented. Suggestions for future research directions related to this thesis are provided in Section 5.2.

5.1 Contributions of the Thesis

5.1.1 Principal Component Analysis of EEG Data

We presented a user-friendly graphical interface to analyze EEG data using multivariate statistical dimensionality-reduction techniques. The proposed approach consists of reading a specified amount of data from an EEG file and conducting linear PCA on the data. We also presented a robust multivariate statistical process control chart using kernel principal component analysis in the analysis of EEGs. The procedure consists of two main steps: First, the data is transformed to a Hilbert feature space using a gaussian kernel. Second, linear PCA is performed on the transformed data. Our experimental results demonstrate the effectiveness of our proposed approach in multivariate dimensionality reduction.

5.1.2 Kernel Isometric Mapping of EEG Data

We presented a robust multivariate statistical process control chart using kernel isometric feature maps (Isomap). The kernel Isomap control chart is based on the eigendecomposition of the Isomap kernel matrix. The proposed control chart is effective in the detection of outliers, and its control limits are derived from the eigen-analysis of the kernel matrix in the Hilbert feature space. Our experimental results demonstrate the effectiveness of using Kernel Isomap methods for dimensionality reduction of multivariate data.

5.1.3 Kernel Locally Linear Embedding Algorithm for Outliers Detection

We presented a new multivariate statistical process control chart for outlier detection using a kernel local linear embedding algorithm. The proposed control chart is effective in the detection of outliers, and its control limits are derived from the eigen-analysis of the kernel matrix in the Hilbert feature space. Our experimental results demonstrate the effectiveness of Kernel LLE control charts in detecting outliers and in the dimensionality reduction of multivariate data.

5.2 Future Research Directions

Several interesting research directions motivated by this thesis are discussed next. In addition to defining additional new control charts based on kernel methods, we intend to perform the following:

5.2.1 Dimensionality Reduction by Principal Variables

Principal variables are a subset of the original variables that preserve the structure and information carried by the original variables, to some extent [25]. The advantage to using principal variables is that the rest of the variables can be discarded. For the EEG data, this should provide a way to determine the EEG channels that contain the best information.

5.2.2 Investigate Additional Dimensionality Reduction Techniques for Control Charts

Hesian Eigenmaps, Laplacian Eigenmaps are a few more dimensionality reduction techniques to be further investigated in the development of new control charts.

List of References

[1] K. Yang and J. Trewn, *Multivariate Statistical Process Control with Industrial Application*, ASA-SIAM, 2002.

[2] D. C. Montgomery, *Introduction to Statistical Quality Control*, John Wiley & Sons, 2005.

[3] L.H. Chiang, E.L. Russell, and R.D. Braatz, *Fault Detection and Diagnosis in Industrial Systems*, Springer, 2001.

[4] J.R. Evans and A. Abarbanel, *Introduction to Quantitive EEG and Neurofeedback*, Academic Press, 1999.

[5] K. Yang and J. Trewn, *Multivariate Statistical Methods in Quality Management*, Mc Graw Hill Professional, 2004.

[6] W. Ku, R.H. Storer, and C. Georgakis, "Disturbance detection and isolation by dynamic principal component analysis," *Chemometrics and Intellignt Labaratory Systems*, vol. 30, pp. 179-196, 1995.

[7] M. Kano, S. Tanaka, S. Hasebe, I. Hashimoto, and H. Ohno, "Combination of independent component analysis and principal component analysis for multivariate statistical process control," *Proc. International Symposium on Design, Operation and Control of Chemical Plants*, pp. 319-324, 2002.

[8] J. Liang and N. Wang, "Fault diagnosis in industrial reheating furnace using principal component analysis," *Proc. IEEE Conference Neural Networks & Signal Processing*, vol. 2, pp. 1615-1618, 2003.

[9] H. Zhang, A.K. Tangirala, and S.L. Shah, "Dynamic process monitoring using multiscale PCA," *Proc. IEEE Canadian Conference on Electrical and Computer Engineering*, vol. 3, pp. 1579-1584, 1999.

[10] H. Jasper, "The ten-twenty electrode system of the International Federation," *Electroencephalography Clinical Neurophysiology*, vol. 10, pp. 371-375, 1958.

[11] "http://homes.esat.kuleuven.be/ biomed/biosource/data007/Interictal_Ictal_EEG.pdf" 2007.

[12] T. Tsagaroulis and A. Ben Hamza, "Statistical process control of electroencephalograms," *Proc. International Conference on Modeling and Simulation*, Montréal, 2007.

[13] W.C. Chin and T.C. Head, *Essentials of Clinical Neurophysiology*, Butterworth-Heinemann, 2002.

[14] I.T. Jolliffe, *Principal Component Analysis*, New York: Springer, 1986.

[15] Y. Bengio, J.F. Paiement, and P. Vincent, "Out-of-Sample extensions for LLE, isomap, MDS, eigenmaps, and spectral clustering," *Technical Report 1238*, Département d'Informatique et Recherche Operational, Université de Montréal, 2003.

[16] T. Tsagaroulis and A. Ben Hamza, "Kernel locally linear embedding algorithm for quality control," *Advances and Innovations in Systems, Computing Sciences and Software Engineering*, Springer, *to appear* 2008.

[17] J.A. Vargas, "Robust estimation in multivariate control charts for individual observations," *Journal of Quality Technology*, vol. 35, no. 4, pp. 367-376, 2003.

[18] N.D. Tracy, J.C. Young, and R.L. Mason, "Multivariate quality control charts for individual observations," *Journal of Quality Technology*, vol. 24, no. 22, pp. 88-95, 1992.

[19] J.H. Sullivan and W.H. Woodall, "A comparison of multivariate control charts for individual observations," *Journal of Quality Technology*, vol. 28, no. 24, pp. 398-408, 1996.

[20] F.A. Alqallaf, K.P. Konis, and R.D. Martin, and R.H. Zamar, "Scalable robust covariance and correlation estimates for data mining," *Proc. ACM International Conference on Knowledge Discovery and Data Mining*, pp. 14-23, 2002.

[21] S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall, 2nd edition, 1998.

[22] S. Roweis and L. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323-2326, 2000.

[23] J. Shawe-Taylor and C. Williams, "The stability of kernel principal components ananlysis and its relation to the process eigenspectrum," *Advances in Neural Information Processing Systems*, vol. 15, 367-374, 2003.

[24] B. Scholkopf, A. Smola, and K-R. Muller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Computation*, vol. 10, pp. 1299-1319, 1998.

[25] J.A. Cumming, D.A. Wooff, "Dimensionality Reduction Via Principal Variables," *Computational Statistics and Data Analysis*, vol. 52, pp. 550-565, 2007.

[26] P. Filzmoser, "A multivariate outlier detection method" *Proc. International Conference on Computer Data Analysis and Modeling*, vol. 1, pp. 18-22, 2004.

[27] M. Hubert, P.J. Rousseeuw, and K.V. Branden, "ROBPCA: a new approach to robust principal component analysis," *Technometrics*, vol. 47, pp. 64-79, 2005.

[28] B.J. Fisch, *EEG Primer: Basic Principles of Digital and Analog EEG*, Elsevier, 1999.

[29] K.H. Chen, D.S. Boning, and R.E. Welch, "Multivariate statistical process control and signature analysis using eigenfactor detection methods," *Proc. Symposium on the Interface of Computer Science and Statistics*, Costa Mesa, CA, June 2001

[30] R.L. Mason, and J.C. Young, *Multivariate Statistical Process Control with Industrial Applications*, ASA-SIAM, 2002.

[31] P. Huber, *Robust Statistics*, John Wiley & Sons, New York, 1981.

[32] F.R. Hampel, E.M. Ronchetti, P.J. Rousseeuw, and W.A. Stahel, *Robust Statistics: The Approach Based on Influence Functions*, Wiley, 1986.

[33] A. Ben Hamza and H. Krim, "Image denoising: a nonlinear robust statistical approach," *IEEE Trans. Signal Processing*, vol. 49, no. 12, pp. 3045-3054, December 2001.

[34] P.J. Rousseeuw and A.M Leroy, *Robust Regression and Outlier Detection*, John Wiley & Sons, New York, NY, 1987.

[35] A.C. Atkinson and H.M. Mulira, "The stalactite plot for the detection of multivariate outliers," *Statistics and Consulting*, vol. 3, pp. 27-35, 1993.

[36] M.S. Chen, J. Han, and P.S. Yu, "Data mining: an overview from a database perspective," *IEEE Transactions on Knowledge and Data Engineering* vol. 8, pp. 866-883, 1996.

[37] U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy. *Advances in Knowledge Discovery and Data Mining*, AAAI/MIT Press, 1996.

[38] P.J. Rousseeuw and K.V. Driessen, "A fast algorithm for the minimum covariance determinant estimator," *Technometrics*, vol. 41, no. 3, pp. 212-223,1999.

[39] W.J. Egan and S.L. Morgan, "Outlier detection in multivariate analytical chemical data," *Analytical Chemistry*, vol. 70, pp. 2372-3279, 1998.

[40] F. Angiulli, S. Basta, and C. Pizzuti. "Distance-based detection and prediction of outliers," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, pp. 145-160, 2006.

[41] R.O. Duda, P.E. Hart, and D.G. Stork, *Pattern classification*, 2nd Edition, Wiley Interscience, 2000.

[42] C.C. Aggarwal and P.S. Yu, "Oulier detection for high dimensional data," *Proc. ACM SIGMOD*, 2001.

[43] F.A. Alqallaf, K.P. Konis, and R.D. Martin, "Scalable robust covariance and correlation estimates for Data Mining," *Proc. ACM SIGKDD*, 2002.

[44] S. Engelen, M. Hubert, and K. Vanden Branden, "A comparison of three procedures for robust PCA in high dimensions," *Austrian Journal of Statistics*, vol. 34, pp. 117-126, 2005.

[45] L.H. Chiang, R.J Pell, and M.B.Seasholtz, "Exploring process data with the use of robust outlier detection algorithms," *Journal of Process Control*, vol. 13, pp. 437-449, 2003.

[46] Z. Ge and Z. Song, "Process monitoring based on independent component analysis-principal component analysis (ICA-PCA) and similarity factors,"*Industrial & Engineering Chemistry Research* , vol. 46, pp. 2054-2063, 2007.

[47] A. Hyvarinen and E. Oja, "Independent component anlysis: algorithms and application," *Neural Networks*, vol. 13, pp. 411-430, 2000.

[48] J.M. Lee, C. Yoo, and I.B Lee, "Statisical process monitoring with independent component anlysis," *Journal of Process Control*, vol. 14, pp. 467-485, 2004.

[49] H. Al-Bazzaz and X.Z. Wang, "New statistical process control chart for Batch operations based on independent component analysis," *Industrial & Engineering Chemistry Research*, vol. 43, pp. 6731-6741, 2004.

[50] J. Mina and C. Verde, "Fault detection using dynamic principal component analysis by average estimation," *Proc. International Conference on Electrical Engineering*, pp. 374-377, 2005.

[51] W. Li and S. J. Qin, "Consistent dynamic PCA based on errors-in-variables subspace identification," *Journal of Process Control*, vol. 11, pp. 661-678, 2001.

[52] J. Chen and C. M. Liao, "Dynamic process fault monitoring based on neural network and PCA," *Journal of Process Control*, vol. 12, pp. 277-289, 2002.

[53] J.F. Cardoso and A. Soulomica, "Blind beamforming for non-Gaussian signals,"*IEEE Proc. Radar & Signal Processing*, vol. 140, pp. 362-370, 1993.

[54] W. Hardle, *Smoothing Techniques*, Springer, 1991.

[55] J.J. Downs and E.F. Vogel, "A plant-wide Industrial process control problem," *Computers & Chemical Engineering*, vol. 17, pp. 245-255, 1993.

[56] M. Kano, K. Nagao, S. Hasebe, I. Hashimoto, H. Ohno, R. Strauss, and B. Bakshi, "Comparison of statistical process monitoring methods: application to the Estman challenge problem," *Computers & Chemical Engineering*, vol. 24, pp. 175-181, 2000.